IN THE UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF TEXAS
MARSHALL DIVISION

IN RE: TAASERA LICENSING LLC,	§	
	§	NO. 2:22-MD-03042-JRG
PATENT LITIGATION	§	

CLAIM CONSTRUCTION MEMORANDUM OPINION AND ORDER

In this multidistrict patent litigation, Taasera Licensing LLC asserts claims from 15 patents generally relating to computer application security against Fortinet, Inc., Palo Alto Networks, Inc., and Musarubra US LLC (together, "Defendants").[1] The parties ask the Court to construe about 25 terms from the patents. Having considered the parties' briefing, along with arguments of counsel during an October 11, 2023 hearing, the Court resolves the disputes as follows.

I. LEGAL STANDARDS

A. Generally

"'[T]he claims of a patent define the invention to which the patentee is entitled the right to exclude.'" *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312 (Fed. Cir. 2005) (enanc) (quoting *Innova/Pure-Water, Inc. v. Safari Water Filtration Sys., Inc.*, 381 F.3d 1111, 1115 (Fed. Cir. 2004)). As such, if the parties dispute the scope of the claims, the court must determine their meaning. *See, e.g.*, *Verizon Servs. Corp. v. Vonage Holdings Corp.*, 503 F.3d 1295, 1317 (Fed. Cir. 2007); *see also Markman v. Westview Instruments, Inc.*, 517 U.S. 370, 390 (1996), *aff'g*, 52 F.3d 967, 976 (Fed. Cir. 1995) (en banc).

Claim construction, however, "is not an obligatory exercise in redundancy." *U.S. Surgical*

[1] Between the claim construction hearing and the issuance of this Order, three cases which were party to this multi-district litigation settled—Case Nos. 2:22-cv-00468-JRG, 2:21-cv-00441-JRG, and 2:22-cv-00063-JRG. The patent claims construed in this Order are currently at issue in this multi-district litigation.

Corp. v. Ethicon, Inc., 103 F.3d 1554, 1568 (Fed. Cir. 1997). Rather, "[c]laim construction is a matter of [resolving] disputed meanings and technical scope, to clarify and when necessary to explain what the patentee covered by the claims . . . ." *Id.* A court need not "repeat or restate every claim term in order to comply with the ruling that claim construction is for the court." *Id.*

When construing claims, "[t]here is a heavy presumption that claim terms are to be given their ordinary and customary meaning." *Aventis Pharm. Inc. v. Amino Chems. Ltd.*, 715 F.3d 1363, 1373 (Fed. Cir. 2013) (citing *Phillips*, 415 F.3d at 1312–13). Courts must therefore "look to the words of the claims themselves . . . to define the scope of the patented invention." *Id.* (citations omitted). "[T]he ordinary and customary meaning of a claim term is the meaning that the term would have to a person of ordinary skill in the art in question at the time of the invention, i.e., as of the effective filing date of the patent application." *Phillips*, 415 F.3d at 1313. This "person of ordinary skill in the art is deemed to read the claim term not only in the context of the particular claim in which the disputed term appears, but in the context of the entire patent, including the specification." *Id.*

Intrinsic evidence is the primary resource for claim construction. *See Power-One, Inc. v. Artesyn Techs., Inc.*, 599 F.3d 1343, 1348 (Fed. Cir. 2010) (citing *Phillips*, 415 F.3d at 1312). For certain claim terms, "the ordinary meaning of claim language as understood by a person of skill in the art may be readily apparent even to lay judges, and claim construction in such cases involves little more than the application of the widely accepted meaning of commonly understood words." *Phillips*, 415 F.3d at 1314; *see also Medrad, Inc. v. MRI Devices Corp.*, 401 F.3d 1313, 1319 (Fed. Cir. 2005) ("We cannot look at the ordinary meaning of the term . . . in a vacuum. Rather, we must look at the ordinary meaning in the context of the written description and the prosecution history."). But for claim terms with less-apparent meanings, courts consider "'those sources available to the

public that show what a person of skill in the art would have understood disputed claim language to mean[,] [including] the words of the claims themselves, the remainder of the specification, the prosecution history, and extrinsic evidence concerning relevant scientific principles, the meaning of technical terms, and the state of the art.'" *Phillips*, 415 F.3d at 1314 (quoting *Innova*, 381 F.3d at 1116).

B. Indefiniteness

"[A] patent is invalid for indefiniteness if its claims, read in light of the specification delineating the patent, and the prosecution history, fail to inform, with reasonable certainty, those skilled in the art about the scope of the invention." *Nautilus, Inc. v. Biosig Instruments, Inc*., 572 U.S. 898, 901 (2014). The claims "must be precise enough to afford clear notice of what is claimed," but that consideration must be made while accounting for the inherent limitations of language. *Id.* at 908.

"Indefiniteness must be proven by clear and convincing evidence." *Sonix Tech. Co. v. Publ'ns Int'l, Ltd.*, 844 F.3d 1370, 1377 (Fed. Cir. 2017). In the context of 35 U.S.C. § 112 ¶ 6, "[t]he party alleging that the specification fails to disclose sufficient corresponding structure must make that showing by clear and convincing evidence." *TecSec, Inc. v. IBM*, 731 F.3d 1336, 1349 (Fed. Cir. 2013) (quoting *Budde v. Harley-Davidson, Inc*., 250 F.3d 1369, 1380–81 (Fed. Cir. 2001)).

II. THE LEVEL OF ORDINARY SKILL IN THE ART

The level of ordinary skill in the art is the skill level of a hypothetical person who is presumed to have known the relevant art at the time of the invention. *In re GPAC*, 57 F.3d 1573, 1579 (Fed. Cir. 1995). In resolving the appropriate level of ordinary skill, courts consider the types of and solutions to problems encountered in the art, the speed of innovation, the sophistication of the

technology, and the education of workers active in the field. *Id.* Importantly, "[a] person of ordinary skill in the art is also a person of ordinary creativity, not an automaton." *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 421 (2007).

Here, Taasera asserts a skilled artisan at the time of the inventions would have had "a bachelor's degree in computer science or computer engineering with one to two years of experience in the field of computer programming for communications systems, or the equivalent education and work experience." Dkt. No. 256 at 2 (citing Cole Decl., Dkt. No. 256-17 ¶¶ 43–45). Defendants criticize this characterization as "overly broad in that it identifies experience in the general field of 'computer programming for communications systems' as sufficient, without requiring specific experience in computer/cyber security." Dkt. No. 259 at 1. Defendants, however, concede any such overbreadth has no bearing on the disputes raised so far. *Id.* Accordingly, the Court adopts Taasera's construction for its analysis of the terms at issue.

III. THE DISPUTED TERMS[2]

2. "if the new program is validated, permitting the new program to continue loading and to execute in connection with the computing device" ('137 Patent, Claims 6, 14, 25)

Taasera's Construction	Defendants' Construction
Plain and ordinary meaning.	If the new program is [validated, claims 6, 13] / [the same as the allowed program, claims 14, 24], then it is not monitored while it [loads and executes in connection with the computing device, claims 6, 13] / [executes on the computing device, claims 14, 24]

[2] Given the number of terms at issue, for easy reference, the Court has kept the numbering for the terms used by the parties.

U.S. Patent 7,637,137 relates to providing "kernel-level protection of a computer system from rogue or malicious computer programs." '137 Patent at 1:17–18. The kernel space is the heart of the operating system and directly accesses the computer's hardware. *Id.* at 5:61–65. In contrast, the user space of the computing device interacts with software and data. *Id.* at 5:65–6:1.

According to the patent, which has a 2002 effective filing date, there were generally two approaches to protecting against malware. '137 Patent at 2:9–10. First, before actual execution of computer code, the device could run the code "virtually" to identify any malicious effect. *Id.* at 2:10–12. This approach is limited because virtual execution performs only a high-level "walk through" rather than execution of every process. *Id.* at 2:17–19. As a result, it tends to frequently detect false positives, thus triggering a high number of security events. *See id.* at 2:19–28.

Second, a monitoring system could be used to control and monitor the device while it executes code. '137 Patent at 2:29–31. Under such an approach, the system monitors active processes and decides whether, and under what conditions, those processes should have access to resources outside the device, such as the internet. *Id.* at 2:49–58. According to the patent, this approach "is not satisfactory because conventional real-time security monitoring solutions do not detect security problems early enough and allow time for a response before the malicious program does harm." *Id.* at 3:1–4.

The patent purports to teach a "protector system" that can "quickly and efficiently examine code in real time, but before it is able to harm a computing device or system." '137 Patent at 3:8–10. First, the system

> determines whether a software program has been previously approved and validates that the software program has not been altered. If the software program is validated during the first phase, this will minimize or eliminate security monitoring operations while the software program is executing during the second phase. If the software program cannot be validated, the protector system enters the second phase and

> detects and observes executing activities at the kernel level of the operating system so that suspicious actions can be anticipated and addressed before they are able to do harm to the computing device.

Id. at [57]. Claim 6 recites an embodiment of the invention as:

> 6. A computer implemented method for implementing security for a computing device comprising the steps of:
>
> interrupting the loading of a new program for operation with the computing device;
>
> validating the new program;
>
> if the new program is validated, permitting the new program to continue loading and to execute in connection with the computing device;
>
> if the new program is not validated, monitoring the new program while it loads and executes in connection with the computing device, wherein the step of monitoring the new program while it executes is performed at the operating system kernel of the computing device.

Id. at 11:37–50; *see also id.* at 12:18–26 (reciting similar language in Claim 14); *id.* at 13:11–17 (reciting similar language in Claim 25).

Defendants present this dispute as "whether a validated program is monitored." Dkt. No. 259 at 3. They contend it is not, pointing to "the '137 Patent's goal of providing 'an effective and efficient method for implementing security while minimizing the burdens and interruptions for the user.'" *Id.* at 3–4 (citing '137 Patent at 10:49–54). In contrast, Taasera stresses the claim language and the patent's disclosure that once a program is validated, "*little or no* additional security monitoring needs to be performed on the new program while it is executed." Dkt. No. 256 at 5 (citing '137 Patent at 3:49–62; emphasis added).

The Court agrees with Taasera. As Defendants' position implicitly acknowledges, the claim language imposes no express restriction on monitoring after a program is validated. Thus, for the

Court to adopt their construction, Defendants must show "clear and unmistakable statements by the patentee that limit the claims, such as 'the present invention includes . . .' or 'the present invention is . . .' or 'all embodiments of the present invention are . . . .'" *Pacing Techs., LLC v. Garmin Int'l, Inc.*, 778 F.3d 1021, 1024 (Fed. Cir. 2015); *see also Chi. Bd. Options Exch., Inc. v. Int'l Sec. Exch., LLC*, 677 F.3d 1361, 1372 (Fed. Cir. 2012) (finding disclaimer when the patent repeatedly disparaged an embodiment as "antiquated" with "inherent inadequacies," and then detailed the "deficiencies [that] make it difficult" to use); *Inpro II Licensing, S.A.R.L. v. T-Mobile USA Inc.*, 450 F.3d 1350, 1354–55 (Fed. Cir. 2008) (finding disclaimer when the specification described a feature as a "very important feature . . . in an aspect of the present invention" and disparaged alternatives to that feature); *SafeTCare Mfg., Inc. v. Tele-Made, Inc.*, 497 F.3d 1262, 1269–70 (Fed. Cir. 2007) (finding disclaimer when the specification indicated the invention operated by "pushing (as opposed to pulling) forces" and characterized the "pushing forces" as "an important feature of the present invention"); *Andersen Corp. v. Fiber Composites, LLC*, 474 F.3d 1361, 1367 (Fed. Cir. 2007) (finding disclaimer when the specification indicated that, for "successful manufacture," a particular step was "require[d]").

Defendants make no such showing here. At best, they point to excerpts characterizing "the present invention" as the two step process of (1) monitoring non-validated software programs before running them to ensure they have not been corrupted, and then (2) monitoring programs that cannot be validated as they execute. *See* Dkt. No. 259 at 4 (citing '137 Patent at 4:53–65, 6:64–67, 9:7–11, 9:38–41). This characterization of "the present invention" says nothing about excluding the monitoring of programs that *have* been validated. At the very least, the patent's characterization that validated programs can (not must) be run without further monitoring removes this from the realm of clear disclaimer. *See* '137 Patent at 3:49–62 ("Once a program is validated in the pre-

execution phase, *little or no additional security monitoring* needs to be performed on the new program while it is executed." (emphasis added)); *id.* at [57] ("If the software program is validated during the first phase, this will *minimize or eliminate* security monitoring operations while the software program is executing during the second phase." (emphasis added)).[3] The Court therefore rejects Defendants' position and gives this term a "plain and ordinary meaning" construction.

3. "an execution module coupled to the detection module and operable for monitoring, at the operating system kernel of the computing device, the program in response to the trigger intercepted by the detection module" ('137 Patent, Claim 1)

Taasera's Construction	Defendants' Construction
Subject to 35 U.S.C. § 112 ¶ 6 **Function:** monitoring, at the operating system kernel of the computing device, the program in response to the trigger intercepted by the detection module	
Structure: FIG. 6.	**Structure:** Not disclosed, so indefinite.

The parties agree this is a means-plus-function term. They also agree on the claimed function, but they disagree about the corresponding structure. Taasera points to Figure 6 (below), Dkt. No. 256 at 6, whereas Defendants say that figure is not linked to the claimed function, Dkt. No. 259 at 6. And even if it is, Figure 6 only discloses more "black boxes." *Id.* At most, say Defendants, steps 640, 645 and possibly step 630 are linked to monitoring. *Id.*

[3] Defendants suggest Taasera's understanding that *some* monitoring can happen after the program is validated introduces a term of degree into the claim, Dkt. No. 259 at 5 n.6, but the issue is whether the claims *exclude* all monitoring—not the amount of monitoring the claims require to find infringement.

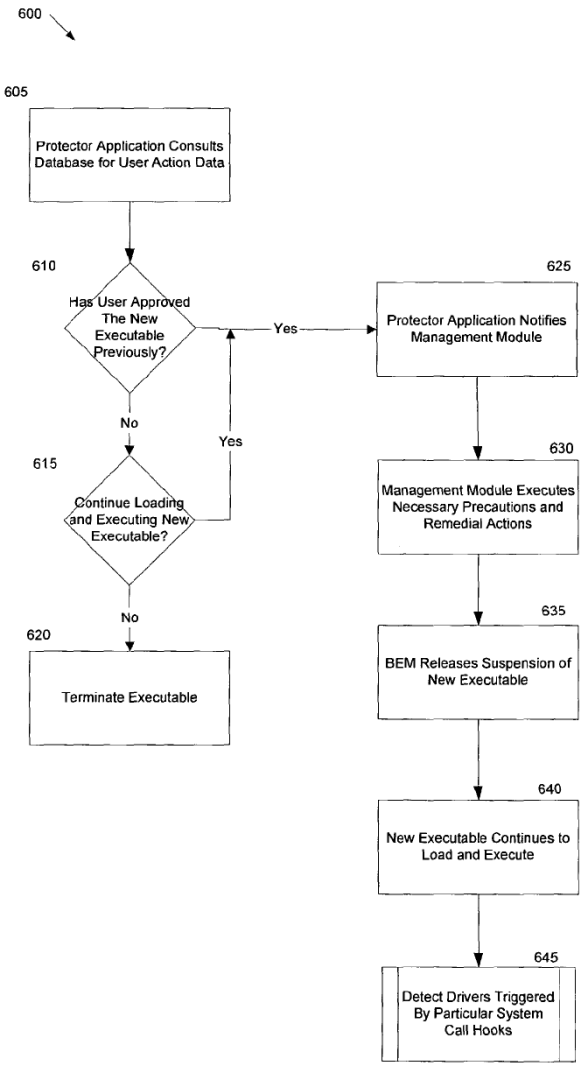

FIG. 6

To start, the Court agrees with Defendants that Figure 6 as a whole does not show an algorithm (or other "structure") for the recited function. At best, step 645 relates to the monitoring function, but the specification only explains "[i]n step 645, certain activities performed by the program will trigger system call hooks that, in turn, trigger the detect drivers 153. The detect drivers are then coupled to the behavior monitoring modules 128, which can observe the program's behavior and respond to any malicious activity." '137 Patent at 10:9–14.

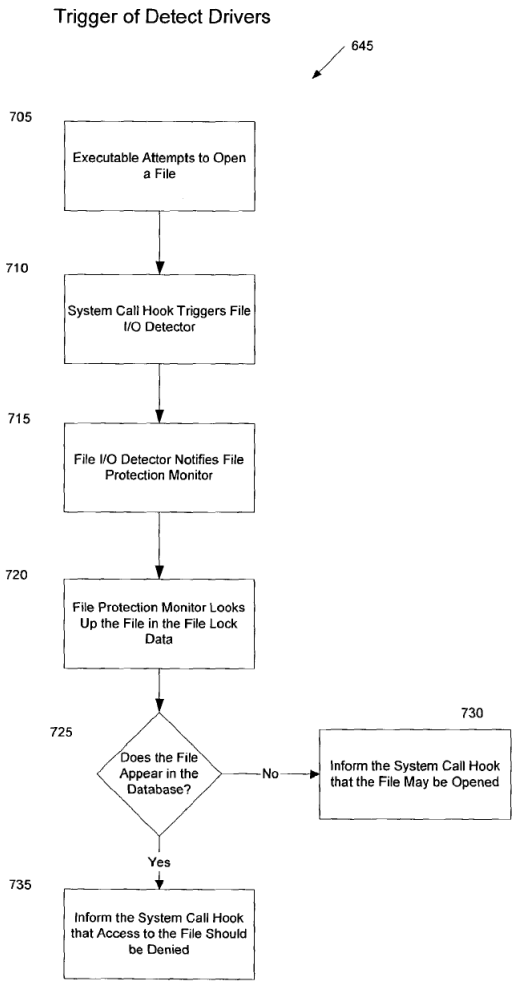

FIG. 7

Figure 7 (above) expounds on step 645 for the file protection monitor and file input/output detector. As the patent explains:

> The exemplary process 645 [of FIG. 6] is illustrative of the operations of detect drivers 153 and their associated behavior monitors 128. In step 705, the executable file attempts to open a file in connection with a process or activity it is performing. As the kernel attempts to follow the instruction of the executable and open the file, the system call hook 175 linked to this activity triggers the file input/output detector 160 in step 710. The file input/output detector 160, in turn, notifies the file protection monitor 135 in step 715.
>
> Any files that have restricted access, as determined by the user or network

> administrator in the setup process 300, will be identified in the file lock data in database 210. The file protection monitor 135 consults the file lock data in step 720 to determine whether the subject file has been restricted. If the file is not restricted, the file protection monitor 135 permits the system call hook 175 to proceed with opening the file. However, if the file does appear in the file lock data 220 in step 735, the file protection monitor can limit access to the file. For instance, the file protection monitor can provide read-only access to a file or can prohibit access entirely.

'137 Patent at 10:17–37.

Although this description of Figure 7 is limited to explaining operation of the file input/output detector 160, it aligns with the specification's description with respect to Figure 1 (below) of the other "detect drivers" monitoring non-validated programs during execution:

> The detect drivers 153 are plug-in modules linked to the system call hooks 175 within the kernel. The detect drivers 153 communicate system call hooks, using component interface 150, to associated behavior monitoring modules 128 that can react to the suspicious activities. The binary execution monitor 125 also works in conjunction with the behavior monitoring modules 128 to analyze and respond to system call hooks communicated from the detect drivers 153. The behavior monitoring modules 128 are a collection of in-kernel modules that are associated with the detect drivers 153. For example, the privacy monitor 130 reacts to a program's attempt to use a network connection to other computer systems. The file protection monitor 135 can react to an attempt to alter a specified file. The registry protection monitor 140 protects against unauthorized changes to registry settings. Generally, the behavior monitors 128 can take direct action to address a security threat or instruct the protector application 115 to query the user for instructions on how to handle the threat. Other behavior monitors 128 and their associated detect drivers 153 can be plugged into the protector system 104 to implement different security functions. . . . The functions of the binary execution monitor 125, the detect drivers 153, and the behavior monitors 128 performed during execution of an executable file can generally be referred to as being performed by an *execution module* of the protector system 104.

'137 Patent at 6:64–7:30. Notably, this is the patent's only reference to an "execution module" in the written description.

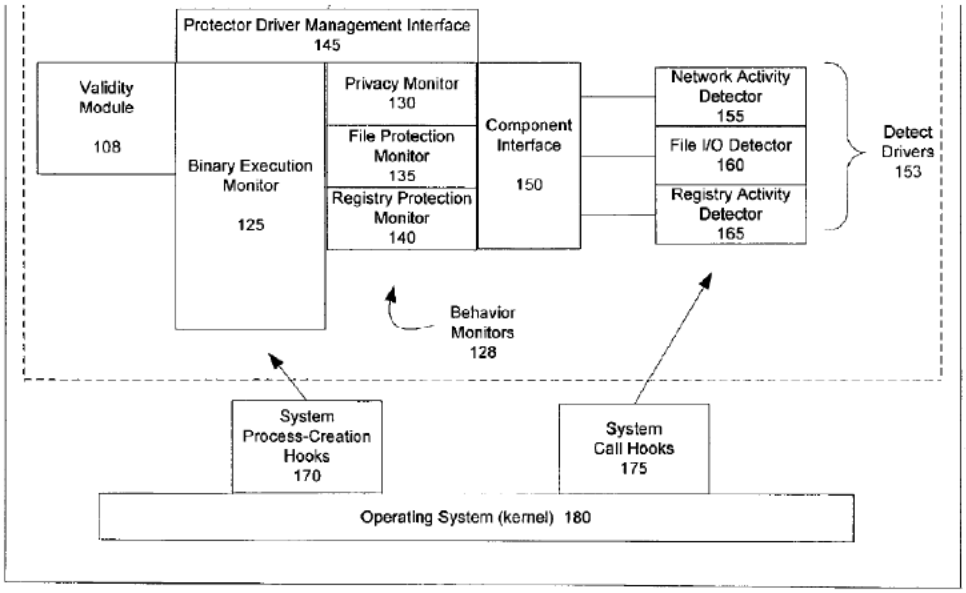

FIG. 1

Reading the paragraph about the "execution module" with Figure 7 explains how the three disclosed behavior protection monitors work. Based on that intrinsic evidence, the Court finds the disclosed algorithm for the claimed monitoring function as: "(1) detecting network activity, an attempt to open a file, or an attempt to change the registry from a system call hook triggered by the kernel; and (2) communicating the system call hook to the associated in-kernel behavior monitor using a component interface."

4. "network administration traffic" ('356 Patent, Claims 1, 2, 9, 10, 13, 14, 17, 18); "[third/fourth] program instructions to determine if the packet is network administration traffic" ('356 Patent, Claims 1, 9, 10, 13, 17)

Term	Taasera's Construction	Defendants' Construction
"network administration traffic	Plain and ordinary meaning.	Indefinite.

"[third/fourth] program instructions to determine if the packet is network administration traffic"	Subject to 35 U.S.C. § 112 ¶ 6. **Structure:** Software algorithm that performs the steps of FIG. 7. **Function:** determine if the packet is network administration traffic.	Subject to 35 U.S.C. § 112(6). **Structure:** Indefinite. **Function:** determine if the packet is network administration traffic.

U.S. Patent 8,127,356 relates to "a technique to detect unknown computer attacks." '356 Patent at 1:8–9. According to the patent:

> Most computer attacks have a characteristic "signature" by which the attack can be identified. The signature can take various forms depending on the nature of the attack, but typically comprises several consecutive lines of plain text or executable code that are distinctive and appear in the attack. Once a signature is determined for a new computer attack, intrusion detection or intrusion prevention software can be created and distributed to customers. The intrusion detection or intrusion prevention software detects the attack from a network interface card (NIC) or when the attack attempts to pass through a firewall. The detection is by a "key word" search for the signature of the attack. The intrusion prevention or intrusion detection software will then thwart the attack by deleting it or preventing its execution by appropriate command to the operating system.

Id. at 1:40–54. The patent then emphasizes the need to identify new exploits quickly so their signatures can be identified and the intrusion prevention or detection software can be created. *Id.* at 1:55–59.

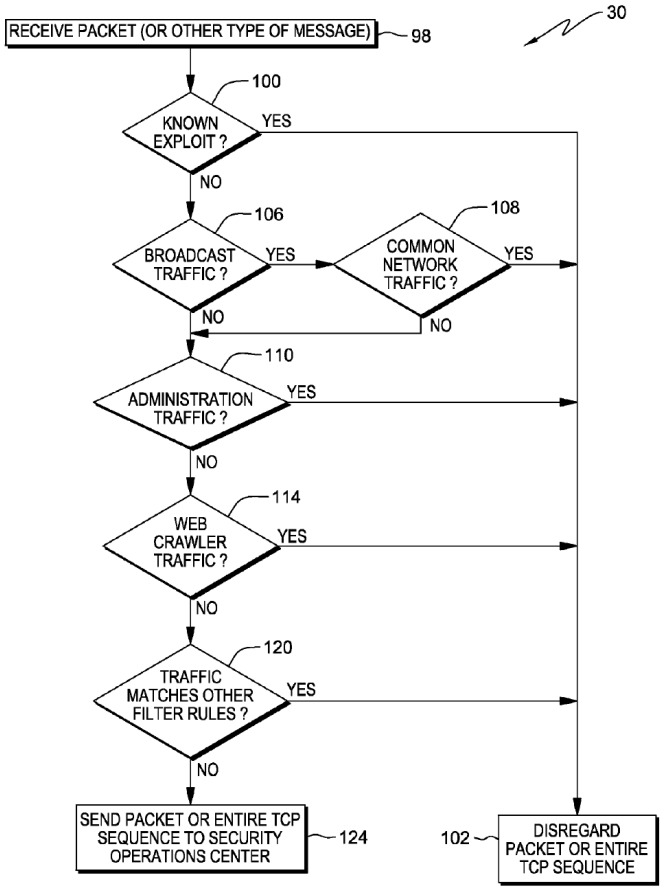

FIG. 2

To help with this identification, the patent teaches a packet filtering program for when the system may disregard a received network packet as a known "exploit." As shown in FIG. 2 (above), the disclosed method determines whether a received network packet is either a known exploit or known form of network traffic, such as broadcast traffic, network administration traffic, or web crawler traffic. If the received packet is not any of these, the program sends the packet to a security operations center ("SOC"), which informs the system operator of an attack by a new exploit. *See generally* '356 Patent at 4:55–5:10. Claim 1 recites an embodiment of the method as:

> 1. A computer program product for automatically determining if a packet is a new, exploit candidate, the computer program product comprising:
>
> a computer-readable tangible storage device;

> first program instructions to determine if the packet is a known exploit;
>
> second program instructions to determine if the packet is addressed to a broadcast IP address of a network;
>
> third program instructions to determine if the packet is network administration traffic;
>
> fourth program instructions, responsive to the packet being a known exploit OR the packet being addressed to a broadcast IP address of a network OR the packet being network administration traffic, to determine that the packet is not a new, exploit candidate; and
>
> fifth program instructions, responsive to the packet not being a known exploit AND the packet not being addressed to a broadcast IP address of a network AND the packet not being network administration traffic AND the packet not being another type of traffic known to be benign, to determine and report that the packet is a new, exploit candidate; and
>
> wherein the first, second, third, fourth and fifth program instructions are stored on the computer-readable tangible storage device.

Id. at 9:31–63.

The parties present two disputes. First, Defendants challenge "network administration traffic" as indefinite for not having a reasonably certain meaning to a skilled artisan. Second, the parties dispute whether the specification discloses sufficient corresponding structure for what they agree is a means-plus-function term.

Given the parties' agreement on both the means-plus-function nature and recited function of "program instructions to determine if the packet is network administration traffic," resolving these disputes is straightforward. According to the patent:

> FIG. 7 [below] illustrates in more detail [the step of] determining if the current packet is network administration traffic presumed to be harmless[]. Some or all bonafide network administrators are known to the administrator of intranet 14 by

> their combinations of IP protocol and respective IP address. These combinations were entered by the administrator and stored in a list 33 within honeypot 12. . . . So, program 30 determines the IP protocol and IP address of the current packet by parsing the packet header (step 500). Then, program 30 compares the combination of IP protocol and IP address of the current packet to the combinations on the list 33 (step 501). If there is a match (decision 502, yes branch), then the current packet is deemed harmless network administration traffic . . . .

'356 Patent at 8:21–37.

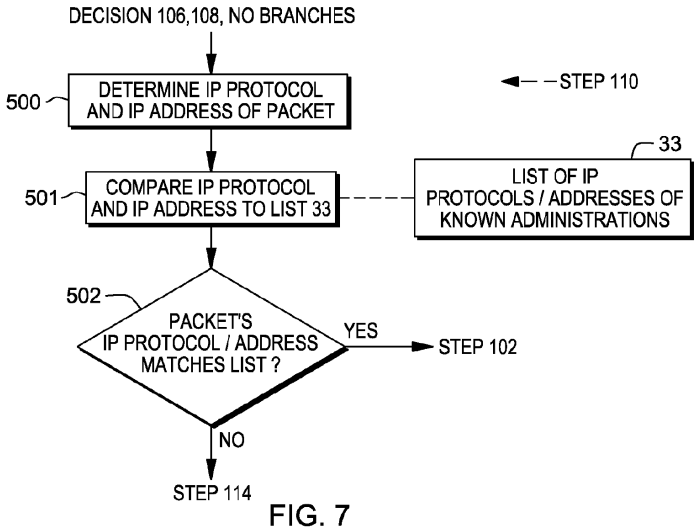

FIG. 7

Based on this description of FIG. 7, the Court adopts the following algorithm as corresponding structure: "(1) determining the IP protocol and IP address of the packet; (2) comparing the determined IP protocol and IP address of the packet to a list of IP protocols and addresses of known administrators; and (3) determining whether the comparison results match."

This also resolves Defendants' challenge to the definiteness of "network administration traffic." That challenge centers on whether "network administration traffic" is characterized by the content of the packet or by whom it was sent. *See* Dkt. No. 259 at 9. But according to the description of FIG. 7, "[i]f there is a match [of IP protocol and IP address of the current packet to the combinations on the list 33], then the current packet is *deemed* harmless network administration traffic . . . ." In other words, content does not matter. The Court therefore construes "network

administration traffic" as "packets having an IP protocol and address that match an IP protocol and address in a list of IP protocols and addresses of known administrators."

5. "attestation" ('441 Patent, Claims 1, 2, 3, 5, 7; '616 Patent, Claim 1)

Taasera's Construction	Defendants' Construction
"verification"	**'441 Patent:** "verifying/verifies the identity of an application" **'616 Patent:** "verifying/verifies the identity of a device, system, or an application" (*see* Dkt. No. 259 at 11)

U.S. Patents 8,327,441 and 9,092,616 relate to "attestation" of the integrity of a computer system or device. Specifically, the '441 Patent relates "to a system and method to provide attestation of applications at runtime." '441 Patent at 1:17–18. Similarly, the '616 Patent relates "to systems and methods for providing dynamic operational integrity attestation of application security and a user reputation at runtime." '616 Patent at 1:16–18.

The claims at issue use "attestation" as a modifier for "service," "server," and "result." For example, Claim 1 of the '441 Patent recites:

> 1. A method of providing *an attestation service* for an application at runtime executing on a computing platform using *an attestation server*, comprising:
>
> receiving, by *the attestation server* remote from the computing platform . . .
>
> a runtime execution context . . . ; and
>
> a security context . . . ;
>
> generating, by *the attestation server*, a report indicating security risks associated with the application based on the received runtime execution context and the received security context, as *an attestation result*; and
>
> sending, by the attestation server, *the attestation result* associated with the application.

'441 Patent at 26:26–45 (emphasis added). Claim 1 of the '616 Patent refers to attestation only in the preamble. '616 Patent at 39:53–55 (reciting "[a] method of providing an attestation service for providing runtime operational integrity of a system using a computing platform").

The parties dispute whether "attestation" only concerns an application's identity, as Defendants propose. Generally, Taasera points to the claims at issue and calls Defendants' construction redundant (for the '441 Patent) and too limiting (for the '616 Patent). Dkt. No. 256 at 12. Defendants cite extrinsic evidence that "attestation" refers to "verifying that a computer is really the computer it claims to be, and is running the software it claims to be running." Dkt. No. 259 at 10 (quoting Dict of Comput. & Internet Terms, Dkt. No. 259-2 at 43).

The Court agrees with Taasera. Nothing about how these patents use "attestation" suggests it must verify the identity of an application. According to Claim 1 of the '414 Patent, the output of the attestation server is an artifact describing the "runtime local execution" or a "security context." '441 Patent at 26:32–39; *see also id.* at 7:13–16 (noting "[a]n attestation service may generate . . . an application statement having at least one of a plurality of attribute value assertions describing the examined runtime local execution and introspection based derived security context"). In fact, Claim 1 requires the output of the attestation to be "a report indicating security risks associated with the application based on the received runtime execution context and the received security context," *id.* at 26:40–43, and does not mention verifying the identity of the application. The '616 Patent expressly relates to operational integrity rather than identity. '616 Patent at 1:16–18. Thus, because the surrounding claim language expressly recites what is being "attested to," the Court rejects Defendants' position and will give this term a "plain and ordinary meaning" construction.

6. "runtime" ('616 Patent, Claim 1)

Taasera's Construction	Defendants' Construction
"[at] the time the system or device is running"	Preamble limiting, at least as to "at runtime"; "[at] the time the application being monitored is running"

Claim 1 recites:

> 1. A method of providing an attestation service for providing *runtime* operational integrity of a system using a computing platform comprising a network trust agent, an endpoint trust agent, and a trust orchestration server, the method comprising:
>
> sending, by the endpoint trust agent on a monitored device, a dynamic context including endpoint events and actions of the monitored device and applications executing on the monitored device at *runtime*; [and]
>
> receiving, at the trust orchestration server, the dynamic context including the endpoint events of the monitored device and the applications executing on the monitored device at *runtime* . . . .

'616 Patent at 39:53–40:9 (emphasis added). The parties dispute whether "runtime" refers only to the time when a monitored application is running.

Defendants contend Taasera's proposed construction renders "runtime" a nullity. Dkt. No. 259 at 12. Specifically, Defendants argue that "[b]ecause the send/receive method steps by definition already require the device to be running, construing 'runtime' to simply mean that the device is running is nonsense." *Id.* Defendants also point to Taasera's prosecution-history statements that require a "live correlation of application actions on a device." *Id.* (citing Dkt. No. 259-4 at 16). According to Taasera, however, "the dynamic context that is sent by the endpoint trust agent and received by the trust orchestration server is of both endpoint events AND applications on the

monitored device at runtime." Dkt. No. 256 at 13–14 (pointing to Claim 1 and citing '616 Patent at 25:65–26:5).

In the context of the claim limitations, the question is whether "at runtime" modifies only "applications executing on the monitored device," or the entire phrase "endpoint events and actions of the monitored device and applications executing on the monitored device." The specification's best language on this issue comes from a description of FIG. 5's "endpoint trust agent," which "can be configured to monitor *all operations* performed on the device 560 at runtime, including running applications and device functions (e.g., microphone, camera, keyboard, or display functions)." '616 Patent 25:66–26:5. In other words, "runtime" is not limited to just applications, but includes device functions, and more broadly, "all operations." That aligns with Taasera's position more than Defendants', because "runtime" refers to when the "system" or "device" is performing any operations, not just specific applications.

Defendants make three arguments for limiting "runtime" to monitored applications, but none of them are persuasive. First, because the system or devices must inherently be running to send or receive information as required by the claims, they say Taasera's construction adds nothing to the claim and renders "at runtime" superfluous. Dkt. No. 259 at 12. Perhaps, but "[t]he preference for giving meaning to all terms . . . is not an inflexible rule that supersedes all other principles of claim construction[.]" *SimpleAir, Inc. v. Sony Ericsson Mobile Commc'ns AB*, 820 F.3d 419, 429 (Fed. Cir. 2016). The Court must still consider how a skilled artisan would have understood the term as used in the specification, and here that understanding weighs against Defendants' position.

Second, Defendants note the parties' agreement on the meaning of "runtime" in claims of other patents at issue. Dkt. No. 259 at 12. They do so, however, without providing any analysis of

why the use of "runtime" in *those* claims is analogous to the use of "runtime" in *this* claim. As Taasera notes in its reply, the other patents have no familial relationship with the '616 Patent.

Finally, Defendants point to prosecution-history remarks in which the applicants distinguished Claim 1 over U.S. Publication No. 2012/0110174 (Wootton). Dkt. No. 259 at 12. Specifically, Defendants cite a statement by applicants that, "in Wootton, there is no live correlation of application actions." *Id.* (citing Amendment Under 37 C.F.R. § 1.114, Dkt. No. 259-4 at 16). But the key distinction in the remarks was that Wootton collected data when the application was *not running*, not that Wootton was limited to *application* runtimes (as opposed to other types of "runtimes"). *See id.* at 15 (describing Wootton as "directed to storing application programs for mobile communications devices in a data store" for analysis; "Wootton describes API analysis, not the analysis of actions of an application executing on a monitored device at runtime").

Based on the specification, the Court construes "runtime" as "the time when operations are performed on the device, including running applications and device functions." Also, because Taasera does not address Defendants' argument about a limiting preamble, the Court holds the preamble of Claim 1 is limiting.

7. "a computing platform comprising a network trust agent" ('616 Patent, Claim 1)

Taasera's Construction	Defendants' Construction
Plain and ordinary meaning.	Limiting preamble; indefinite.

Claim 1 is directed to "[a] method of providing an attestation service for providing runtime operational integrity of a system using a computing platform comprising a network trust agent, an endpoint trust agent, and a trust orchestration server[.]" '616 Patent at 39:53–56. According to Defendants, the context of the claim does not explain what the "network trust agent" is, and the

phrase was not a well-known term of art on the effective filing date. Dkt. No. 259 at 13. Taasera, however, suggests "network" merely modifies "trust agent" and criticizes Defendants' expert for a "feigned lack of understanding" about the term's meaning. Dkt. No. 256 at 14.

To start, the Court disagrees with Taasera that "network" merely modifies "trust agent" and the inquiry ends there. That argument suggests the "trust agent" at the end point and network are identical, but the specification shows otherwise. For example, the *endpoint* trust agent of FIG. 5 comprises (1) "a process monitor configured to observe local execution context of applications and services," (2) "a socket monitor configured to observe network activities of applications and services" (3) "a system monitor configured to observe system and platform resources consumed by applications and services"; (4) "an application integrity module"; and (5) "a resource utilization module configured to assess operational integrity based on rulesets." *See* '611 Patent 13:4–15 (reference numbers omitted). Elsewhere, the patent discloses "[t]he endpoint trust agent sends a dynamic context that may include the local runtime execution context of a mobile application to the trust orchestrator that may perform a calculus of risk on a global security context, that may include endpoint assessment reports received from collaboration services, and sends threat intelligence as a subscription based reputation service to a network trust agent *for network flow remediation*." *Id.* at 26:20–28 (emphasis added; reference numbers omitted).

In contrast, the *network* trust agent (1) "sends messages to the network security framework middleware to send policies to a wireless access point, or send policies to a wireless access point 1320, to apply access controls to block or divert traffic flows," '616 Patent at 26:29–34 (reference numbers omitted); (2) "may generate and send to the computer network protocol (e.g., OpenFlow™) security framework directives to formulate commands for the computer network protocol (e.g. OpenFlow™) controller," *id.* at 33:10–14 (reference numbers omitted); and (3) "generates

and sends to the wireless access point policies to apply access controls for the mobile device," *id.* at 34:5–8 (reference numbers omitted).

Given the fundamental differences in how the specification describes the two "trust agents," and in light of Defendants' unrebutted expert testimony that the term does not have a well-understood meaning, Taasera's "plain meaning" construction is not sufficient. However, it is not indefinite in light of the specification. The Court construes the term as "software that sends policies to a network access point to block or divert traffic flows for user devices based on system warnings associated with those devices." *See* '616 Patent at 26:29–34. Additionally, the Court finds the pre-amble limiting given Taasera's agreement on that issue. *See* Hr'g Tr., Dkt. No. 305 at 33:7–8.

8. "at runtime receiving . . . a runtime execution context indicating attributes of the application at runtime, wherein the attributes comprise one or more executable file binaries of the application and loaded components of the application" ('441 Patent, Claims 1, 4)

Taasera's Construction	Defendants' Construction
Plain and ordinary meaning subject to the construction of "runtime."	"Receiving at the time the relevant program is running an execution context that includes the executable file binaries of the application (as distinct from binary hashes) and loaded components of the application."

The crux of the parties' dispute is whether "binary hashes" are different from "executable file binaries," and thus would fail to meet this limitation. *See* Hr'g Tr., Dkt. No. 301 at 48:6–7 (characterizing "the executables being not binary hashes [as] the most important part of [Defendants'] proposed construction"). Defendants assert the specification treats hashes and executable files differently. Dkt. No. 259 at 15 (citing '441 Patent at 6:7–8, 8:8–9, 8:18–16, 11:44–45, 11:50–52). They also point to the prosecution history as distinguishing between the two. *Id.* at 15–16. Taasera's briefing focuses mainly on Defendants' prosecution-history argument and does not explain the difference between hashes and "executable file binaries." *See* Dkt. No. 256 at 16–18; Dkt. No. 272 at 6.

The Court agrees with Defendants. The specification confirms a "hash" is an attribute of the application rather than the application itself. *See* '441 Patent at 6:5–14 (differentiating between an application and "a hash of the application, as an application fingerprint"). Taasera acknowledges as much. *See* Hr'g Tr., Dkt. No. 301 at 41:10–14. ("[A] binary hash is kind of like a fingerprint of a file. So a hash is when you take a larger file that may be binary, 1s and 0s, and you perform some algorithm on it that turns it into a smaller shorthand for that file that you can use to identify it."); *see also id.* at 44:13–17 ("The idea is to have a shorthand that you can easily show or list that can

be compared. And that hash will be used as an identifier for or an attribute of that file. So a binary hash of an executable is an attribute of that execut[able] file, like its fingerprint."). Thus, the parties do not appear to dispute that the ordinary meaning of "executable file binaries" does not include "binary hashes," and the Court agrees. Thus, a "binary hash" does not literally meet the "executable file binary" requirement of these claims. The Court otherwise adopts a "plain and ordinary meaning" for this phrase.

9. "a security context providing security information about the application" ('441 Patent, Claims 1, 4, 5)

Taasera's Construction	Defendants' Construction
Plain and ordinary meaning.	"security information about the application provided by a collaboration service"

Claim 1 requires "an attestation server remote from the computing platform" to receive "a security context providing security information about the application, wherein the information comprises an execution analysis of the one or more executable file binaries and the loaded components." '441 Patent at 26:27–45. Defendants assert that, although the claims recite what the "security context" *does*, they say nothing about what the security context *is*. Dkt. No. 259 at 16. Taasera accuses Defendants of attempting to import a limitation into the claim with its construction. Dkt. No. 256 at 18.

The Court agrees with Taasera. To address the alleged uncertainty about what a "security context" *is*, Defendants' construction limits from *where* the security context comes. The Court is not persuaded the source of the security context is a defining attribute. Generally, "context" refers to what is known about the application based on attributes. Thus, "security context" would be what information known about the application security based on the attributes of the "executable file binaries and the loaded components." Defendants have not shown the term's meaning depends on

what system component provides it. Accordingly, the Court rejects Defendants' position but will otherwise give this term a "plain and ordinary meaning" construction.

10. "an application artifact" ('441 Patent, Claim 2)

Taasera's Construction	Defendants' Construction
Plain and ordinary meaning	"data identifying one or more attribute value assertions that describe the application runtime execution context"

Claim 2 recites "generating, by the attestation server, *an application artifact* as a reference for changes in a subsequent execution context." '441 Patent at 26:46–52 (emphasis added). Taasera criticizes Defendants' construction as improperly narrowing the limitation to data that "identifies one or more attribute value assertions." Dkt. No. 256 at 20–21 (citing '441 Patent at 7:27–32).

Defendants' construction is rooted in the Abstract, which generally describes the invention as having an attestation service that "may generate an application artifact having associated therewith a name and an application statement having at least one of a plurality of attribute value assertions describing the examined runtime local execution and introspection based derived security context." '441 Patent at [57]. Elsewhere, the specification refers to using the "application artifact" as a reference for determining whether the application has changed. *See, e.g.*, *id.* at 7:27–32 ("The attestation broker may issue application artifacts (e.g., that may maintain a record of the state of the discovered or identified applications running on the instrumented platform) to the runtime monitor for discovered or identified applications running on the instrumented platform." (reference numbers omitted). These excerpts support Defendants' construction.

Taasera calls Defendants' construction too narrow for two reasons. First, says Taasera, Claim 2 defines an application artifact as a reference for change in a subsequent execution context. Dkt. No. 272 at 7. That, however, is not inconsistent with Defendants' construction, and simply

limits the purpose of the artifact so far as Claim 2 is concerned. Second, the specification shows application artifacts can be "record(s) of the state of the discovered or identified applications running on the instrumented platform 100)," Dkt. No. 272 at 7 (citing '441 Patent at 7:27–32). Here, too, the passage on which Taasera relies provides a purpose of the "application artifacts"—"maintaining a record of the state" of an application—rather than a characteristic of them. The Court adopts Defendants' construction for this term.

11. "introspective security context" ('441 Patent, Claims 4, 5)

Taasera's Construction	Defendants' Construction
Plain and ordinary meaning.	"a security context based on evaluation of historic state information and measurements [of the remote computing platform] sampled over a period of time" Otherwise, indefinite.

Claim 1 of the '441 Patent recites receiving, by an attestation server, "a security context providing security information about the application, wherein the security information comprises an execution analysis of the one or more executable file binaries and the loaded components." '441 Patent at 26:30–37. Claim 4 then limits the received security context to "an introspective security context." *Id.* at 26:57–61. The parties dispute the effect of "introspective" as a modifier of "security context."

The parties agree on the relevant portion of the specification:

> The attestation broker may request *introspection based security context* (for example, evaluations and behavioral or predictive analytics based on historic state information and measurements sampled over a period of time using a variety of inspection methods) for the running application on the instrumented platform from one or more of the plurality of collaboration services.
>
> In certain exemplary embodiments, the collaboration services may perform just-in-time inspection (e.g., an assessment scan) of the target application and platform. The collaboration service may lookup the most recent inspection report based on,

> for example, an IT service management schedule and may return the requested attestations pertaining to the application execution context to the requestor or the user.

'441 Patent at 9:40–54 (reference numbers omitted). Defendants base their construction on the first paragraph, whereas Taasera says the proper construction must allow for the substance of the second paragraph. Taasera accuses Defendants of reading examples into the claims, but does not provide an "ordinary meaning" of the term for consideration. Dkt. No. 256 at 21.

The Court agrees with Defendants. "Introspection" suggests looking "inward." Despite the first paragraph's use of "for example" within the parenthetical, the use of internal information and measurements over time is best explains the effect of the "introspective" modifier. Taasera's opening brief provides no basis for the Court to conclude otherwise, suggests no "plain meaning" of its own, nor asserts any lexicography or disclaimer. Accordingly, the Court construes "introspective security context" as "a security evaluation based on information and measurements [of the remote computing platform] sampled over a period of time."

12. "the application of the restriction of the user's transaction" ('441 Patent, Claim 11)

Taasera's Construction	Defendants' Construction
Plain and ordinary meaning.	Indefinite.

Claims 9–11 of the '441 Patent recite:

> 9. The method according to claim 1, further comprising providing confidence metrics in the attestation results indicating a level of security risk by different classifications such that *a restriction on a user's transaction with the application* are applied based on the level of security risk indicated by the confidence metrics in the attestation results.
>
> 10. The method according to claim 9, wherein *the application of the restriction on the user's transaction* includes applying the restriction on the user's network access to the application.

> 11. The method according to claim 1, wherein *the application of the restriction on the user's transaction* includes applying routing decisions and redirecting the user to an alternate computer platform.

'441 Patent at 27:12–25 (emphasis added).

Defendants assert Claim 11 is indefinite because Claim 1 provides no antecedent basis for the disputed term. Dkt. No. 259 at 20. They recognize Claim 11 could properly depend from either Claim 9 or 10, but argue a skilled artisan would have no way of discerning the proper dependency. *Id.* at 20–21. Taasera, however, contends this is a correctable clerical error, and the specification shows Claims 10 and 11 are directed to different embodiments and should each depend from Claim 9. Dkt. No. 256 at 23. It points to two paragraphs allegedly showing the error is not subject to reasonable debate:

> In certain exemplary embodiments, a user's transaction with the application may be controlled by applying a set of authorization rules in accordance with the attestation results. In these and other exemplary embodiments, a *user's network access to the application may be controlled based on the set of authorization rules and the attestation results*.
>
> . . .
>
> In certain exemplary embodiments, the restriction on the user's transaction may include the application of routing decisions and the redirection of the user to an alternate computer platform.

Dkt. No. 256 at 23 (quoting '441 Patent at 19:25–40; emphasis by Taasera).

"A district court may correct 'obvious minor typographical and clerical errors in patents.'" *Pavo Solutions LLC v. Kingston Tech. Co.*, 35 F.4th 1367, 1373 (Fed. Cir. 2022) (quoting *Novo Indus., L.P. v. Micro Molds Corp.*, 350 F.3d 1348, 1357 (Fed. Cir. 2003)). "Correction is appropriate 'only if (1) the correction is not subject to reasonable debate based on consideration of the claim language and the specification and (2) the prosecution history does not suggest a different

interpretation of the claims.'" *Id.* (quoting *Novo Indus.*, 350 F.3d at 1354). "The error must be 'evident from the face of the patent' . . . and the determination 'must be made from the point of view of one skilled in the art[.]'" *Id.* (quoting *Grp. One, Ltd. v. Hallmark Cards, Inc.*, 407 F.3d 1297, 1303 (Fed. Cir. 2005), and *Ultimax Cement Mfg. Corp. v. CTS Cement Mfg. Corp.*, 587 F.3d 1339, 1353 (Fed. Cir. 2009)).

Here, Taasera's proposed correction is subject to reasonable debate. The excerpts to which Taasera points are not inconsistent with Claim 11 depending from *either* Claim 9 or Claim 10. And although the specification does not disclose an embodiment that includes the limitations of both Claim 10 and Claim 11, claims can cover more than just disclosed embodiments. Because Claim 11 could properly depend from either Claim 9 or Claim 10, the correction is subject to reasonable debate. Accordingly, the Court declines to correct the "error" and holds Claim 11 indefinite.

13. "return URL" ('419 Patent, Claims 1, 3, 4, 6, 7, 9, 10, 12, 13, 15, 16, 19; '634 Patent, Claims 1, 3, 4, 5, 6, 8; '251 Patent, Claims 1, 2, 3, 4, 5, 6, 8, 9, 10; '453 Patent, Claims 1, 5, 6, 10, 11, 12, 16, 17, 18)

Taasera's Construction	Defendants' Construction
Plain and ordinary meaning.	"a new URL for the requested resource that is returned to the requestor's web browser"

These related patents teach "a method and system for protecting computing systems from unauthorized access, gained through the contents of URLs, by dynamic encryption of URLs." *See* '419 Patent at 1:10–13. Specifically, the patents teach

> protect[ing] access to computing systems via a URL by encrypting all or a portion of the URL during the transmission of information over a network. Prior to the actual transmission of the information, fields in the URL can be encrypted using conventional encryption techniques. The encryption will occur after the destination has been determined such that the encryption will not cause the information to be

> misdirected to a wrong destination. At the destination location, there is first a determination that this URL is an encrypted URL. At this point, a decryption technique is employed based on the predetermined encryption scheme to decrypt the URL. The full URL is now accessed only by the destination location and is not accessible during travel over the network between the originating and destination locations.

'419 Patent at [57]. Claim 1 of the '419 Patent recites an embodiment of a method as:

> 1. A method for restricting access to information transmitted over a computing network, said method comprising the steps of:
>
> receiving, by a computer at a network location, a resource request for a resource to be located, said resource request containing a universal resource locator (URL);
>
> evaluating, by the computer, the URL to determine whether encryption of none, part, or all of the URL is required;
>
> determining by the computer, whether the requested resource is available;
>
> locating, by the computer, the requested resource contained in the resource request, when the determination is that the requested resource is available;
>
> encrypting, the computer, the URL contained in the resource request; and
>
> determining, by the computer, whether encryption is required for none, part, or all of a *return URL* of the requested resource that is to be returned to a location of the resource request.

Id. at 9:31–49 (emphasis added).

The dispute concerns the scope of "return URL." Arguing "return URL" would not have an understandable meaning to a jury, Defendants contend the "return URL" is a "new URL." Dkt. No. 259 at 35. Their position hinges on the disclosure that, "if the URL must be encrypted, . . . the encrypted URL value is calculated" and "[t]he new, encrypted URL is returned to the browser via a redirect procedure." *Id.* (quoting '441 Patent at 6:66–7:6). According to Taasera, Defendants'

construction "directly contradicts the patent specification, which notes the requested resource can be either returned with or without a URL change[.]" Dkt. No. 256 at 24 (citing '419 Patent at 6:64–7:4).

The Court agrees with Taasera. The surrounding claim language explains the proper meaning of the term—the URL "that is to be returned to [the] location of the resource request." '419 Patent at 9:46–49 (Claim 1); *see also, e.g.*, *id.* at 11:5–7 (reciting, in Claim 10, "instructions for determining whether [to encrypt] none, part, or all of a return URL of the requested resource *that is to be returned to a location of the resource request*" (emphasis added)); '251 Patent at 10:1–5 (reciting, in Claim 1, "determining, by the computer, whether encryption is required for a return URL of the requested resource *that is to be returned to a location of the resource request*" (emphasis added)). Thus, the "return URL" may or may not be "new" or "encrypted." The Court therefore rejects Defendants' position that this term would not have an understandable meaning to a jury and will give this term a "plain and ordinary meaning" construction.

14. "evaluating[, by the computer,] the URL to determine whether encryption of [none, part, or all of] the URL is required" ('419 Patent, Claims 1, 4, 10, 13, 17; '634 Patent Claims 1, 4)

"determining, by the computer, whether encryption is required for none, part, or all of a return URL" / "determining[, by the computer,] [whether/that] encryption of [a/the] return URL [of the requested resource] is required" / "determining by the computer, [whether/that] encryption of the contained URL [is/is not] required" / "determine that encryption of the URL is not required" ('419 Patent, Claims 1, 4, 13, 19; '634 Patent, Claims 1, 4; '251 Patent, Claims 1–6, 8–10; '453 Patent, Claims 1, 4, 6–18)

Term	Taasera's Construction	Fortinet's Construction

"evaluating[, by the computer,] the URL to determine whether encryption of [none, part, or all of] the URL is required"	Plain and ordinary meaning.	"deciding[, by the computer,] whether encryption should be performed on [none, part, or all] of the URL"
"determining, by the computer, whether encryption is required for none, part, or all of a return URL" "determining[, by the computer,] [whether/that] encryption of [a/the] return URL [of the requested resource] is required" "determining by the computer, [whether/that] encryption of the contained URL [is/is not] required" "determine that encryption of the URL is not required"	Plain and ordinary meaning.	"deciding[, by the computer,] [whether / that] encryption should be performed on [none, part, or all of] a return URL" "deciding[, by the computer,] whether encryption should be performed on [a/the] return URL [of the requested resource]" "deciding, by the computer, [whether/that] encryption [should / should not] be performed on the contained URL" "deciding[, by the computer,] that encryption should not be performed on the [return] URL"

According to Fortinet, Taasera's infringement contentions suggest these limitations can be literally met "when a computer simply looks at a URL it has received to determine whether the URL is already encrypted." Dkt. No. 259 at 37. It argues "simply observing whether the URL [is] already encrypted before it was received" ignores that the invention concerns unencrypted URLs, and already-encrypted URLs do not need the patents' teachings. *Id.* The Court agrees.

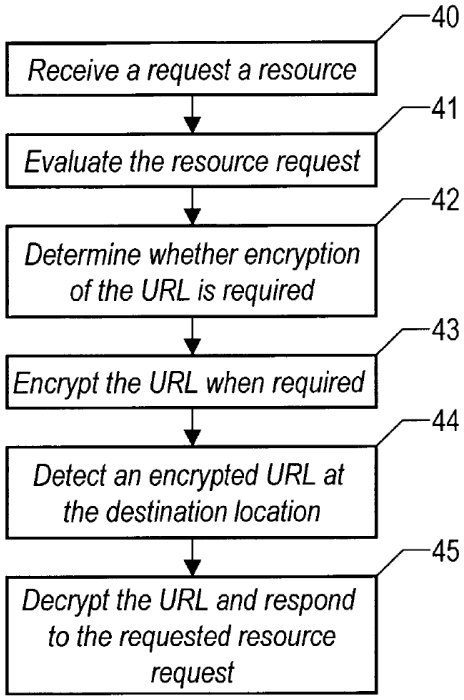

FIG. 2 of the '419 Patent

With respect to FIG. 2 (above), the patent explains:

> [S]tep 41 evaluates the URL of the originating request to determine the location of the requested resource and any contents of the message. *Based on information, which could include the location of the requested resource and the contents of the originating request message, step 42 determines whether the URL of the original resource request requires encryption prior to transmission of the request.* . . . If the URL requires encryption, step 43 encrypts the URL.

'419 Patent at 6:14–24. The method makes a similar determination when returning the page to the browser. *Id.* at 7:45–48 ("If the page is accessible, there is a determination whether this page should be returned to browser in an encrypted form, step 71. This determination can be based on the security policy established for that system.").

Taasera contends Fortinet improperly makes an infringement argument that should be ignored, Dkt. No. 272 at 9, but the Court disagrees. To the contrary, Fortinet argues the scope of the

disputed phrases excludes Taasera's interpretation—that is, Taasera's "construction"—of the terms as applied in its infringement contentions. Whether that exclusion is proper is squarely within the realm of claim construction and separate from the later step of comparing the accused device to the properly construed claims.

Regardless, Taasera appears to agree with Fortinet's position, stating "the claim language in each of the four patents is clear enough that the 'determining' or 'evaluating' is not 'checking' if the URL is encrypted already." Dkt. No. 272 at 9. As such, the Court will give these terms a "plain and ordinary meaning" construction, but for clarity expressly rejects that simply observing whether the URL is already encrypted before it is received falls within the scope of these terms.

16. "determining[, by the computer,] whether the URL of the requested resource is required" ('419 Patent, Claims 2, 11; '634 Patent, Claim 2)

Taasera's Construction	Fortinet's Construction
Plain and ordinary meaning.	Indefinite.

As noted *supra*, Claim 1 of the '419 Patent recites the step of "determining by [a] computer [at a network location], whether [a] requested resource is available." '419 Patent at 9:39–40. Claim 2 then recites, if the requested resource is available and encryption of the requested resource is required, "*determining, by the computer, whether the URL of the requested resource is required*." *Id.* at 9:53–59 (emphasis added); *see also id.* at 11:13–15 (requiring, in Claim 11, "instructions for determining whether the URL of the requested resource is required when encryption of the requested resource is required"); '634 Patent at 9:51–54 (reciting, in Claim 2, the step of "determining, by the computer, that encryption of the requested resource is required and in response, determining, by the computer, whether the URL of the requested resource is required").

The parties dispute whether "determining, by the computer, whether the URL of the

requested resource is required" is indefinite. Fortinet calls this language incoherent with no plain meaning. Dkt. No. 259 at 40. Taasera, on the other hand, argues this term "reflects the computer further determining whether to include the URL along with the requested resource to the destination . . . that requested the resource." Dkt. No. 256 at 29. According to Taasera, "[t]he specification is clear that a URL may or may not be required along with the requested resource." *Id.* (citing '419 Patent at 7:9–18).

The Court disagrees with both parties. To start, the excerpt to which Taasera cites says nothing about whether a URL is required. Rather, it describes conditions that trigger an error message returned to a browser requesting a resource. *See* '419 Patent at 7:9–18. Moreover, Taasera's position makes no sense in the context of the disclosure, which is about encrypting URLs to protect against unauthorized access, not whether URLs are "required."

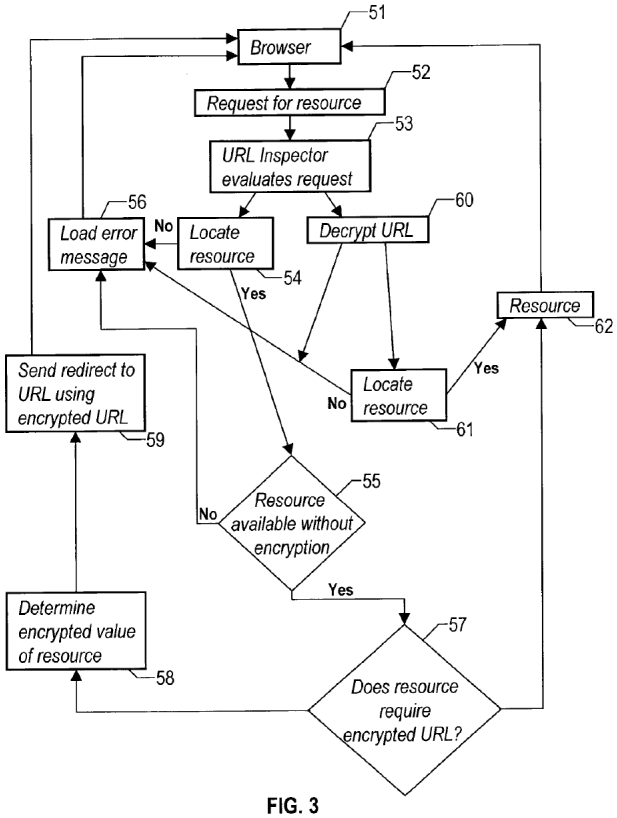

FIG. 3

A better interpretation comes from the discussion of FIG. 3 (above), which closely tracks the claim language. The patent explains:

> In the case of block 55, since this resource has been requested using a plain text URL, the system must determine if the resource is available without using an encrypted URL (i.e. just because it is there, it might not be accessible using a plain text URL). If the resource is not available without encryption, the process moves to block 56. If the resource *is available* without encryption, *there must be a determination whether the resource URL requires encryption before returning the resource to the browser of the original requester.*

'419 Patent at 6:54–63 (emphasis added). In other words, the determination is not whether the URL of the requested resource is required, but rather whether *encryption* of that URL is required. Fortinet recognizes at least that possibility in its response. *See* Dkt. No. 259 at 41 (calling a construction of "determining . . . whether encryption of the URL . . . is required" a "possible correction").

The Court recognizes this language from the specification does not track the claim language precisely. The specification describes the "determining" step as occurring "if the resource is available *without* encryption," whereas the "determining" step of the claims happens "when encryption of the requested resource *is* required." Nonetheless, given the similarity of the language and the concepts involved, it shows the proper construction: "determining[, by the computer,] whether *encryption of* the URL of the requested resource is required."

17. "compliance state of the endpoint" ('038 Patent, Claims 1, 12, 23; '997 Patent, Claims 1, 11, 21; '918 Patent, Claims 1, 9, 17)

Taasera's Construction	Defendants' Construction
Plain and ordinary meaning.	"level of compliance by the endpoint with compliance policy thresholds"

These patents, which are related and have the same disclosure, concern "controlling access

to computing resources based on known computing security vulnerabilities." '918 Patent at 1:29–30. Generally, the patents describe the inventions as methods and systems

> for fine tuning access control by remote, endpoint systems to host systems. Multiple conditions/states of one or both of the endpoint and host systems are monitored, collected and fed to an analysis engine. Using one or more of many different flexible, adaptable models and algorithms, an analysis engine analyzes the status of the conditions and makes decisions in accordance with pre-established policies and rules regarding the security of the endpoint and host system. Based upon the conditions, the policies, and the analytical results, actions are initiated regarding security and access matters.

Id. at [57].

Each of the claims at issue refers to both a "compliance state" of an endpoint and a "compliance policy" (or "compliance policies"). For example, Claim 1 of the '918 Patent recites:

> 1. A method for controlling the operation of an endpoint, comprising:
>
> . . .
>
> determining, by the computing system, a *compliance state* of the endpoint based on the user information and status information, and a plurality of *compliance policies* in the data store;
>
> authorizing access by the endpoint to a computing resource on the network, authorization being determined by the remote computing system in response to the *compliance state*; and
>
> continuing to monitor the *compliance state* by the endpoint and restricting access to the computing resource if the *compliance state* changes.

'918 Patent at 61:29–56 (emphasis added); *see also* '038 Patent at 60:34–55 (reciting, in Claim 1 "[a] method for controlling the operation of an endpoint" comprising the steps of "determining, by the computing system, *a compliance state* of the endpoint based on the status information and *a plurality of compliance policies* in the data store" and "initiating, by the computing system, based

on *the compliance state*, an action identified in at least one rule in the data store" (emphasis added)).

Defendants base their construction of "compliance state" on excerpts from the specification that explain "the compliance state of the endpoint results from comparing data about multiple endpoint conditions (*e.g.,* in the form of 'compliance scores'), with policy defined thresholds from a compliance policy." Dkt. No. 259 at 43 (citing '038 Patent at 39:16–17 ("These compliance scores are compared to policy-defined thresholds in order to make a compliance assessment."); *id.* at 9:29–33 ("[T]here are many different data sources and data elements that can be examined to assess the state of the endpoint, form compliance assessments, and ultimately make policy-based access control decisions regarding local and remote computing resources.")). From these excerpts, Defendants conclude a "compliance state" must be a "level" of compliance. Dkt. No. 259 at 43.

That conclusion improperly excludes the possibility the state could simply be "in compliance" or "not in compliance," without any "levels." For example, the specification teaches an "antivirus compliance policy" where the system is compliant if the "most recent antivirus state inspection has a compliance score greater than 50 OR a mean of [the] past 5 consecutive samples [is] greater than 70." '038 Patent at 50:4–8. Otherwise, the system is not in compliance. In fact, the notion of being "in compliance" or "not in compliance" appears frequently in the specification. *See, e.g., id.* at 39:16–29 (disclosing a situation in which an endpoint is "out of compliance with regards to currently running security agents and their vendor [and] in compliance with regards to current configuration settings"); *id.* at 51:4–9 (noting "policy actions may be endpoint actions allowed to take place because the endpoint system 104 is *in compliance with security policies*, actions to take to partially or wholly restrict access to endpoint resources because the endpoint system 104 is *not in compliance with security policies*"). In contrast, none of the excerpts on which

Defendants rely require "levels." At most, they explain how the compliance state is determined. The Court therefore construes "compliance state" as "the state of being either in or not in compliance."

18. "compliance polic[y/ies]" ('038 Patent, Claims 1, 12, 23; '997 Patent, Claims 1, 11, 21; '918 Patent, Claims 1, 9, 17)

Taasera's Construction	Defendants' Construction
Plain and ordinary meaning.	"The items on an endpoint to monitor, the analysis methods to use, and the permitted thresholds for the monitored items"

This dispute relates to the specification's explanation that "[d]ifferent sets of compliance policies may have the same or different values regarding items monitored, compliance threshold, analysis methods to use, etc." Dkt. No. 259 at 45 (citing '038 Patent at 56:57–62). From this excerpt, Defendants conclude a "compliance policy" must include all three of those characteristics.

The Court sees no basis for limiting the scope of the term that way. Even if all of the disclosed "compliance policies" meet Defendants' construction, that does not warrant changing the term's ordinary meaning. Here, for example, a simple policy for allowing an endpoint attempting to access a network might be to require encryption.[4] In that case, there would be no items to monitor or permitted thresholds—it would be a straightforward question of, "Is the device using encryption?" The Court sees no reason this would not be a "compliance policy" within the scope of the claims or why Defendants' narrow construction is warranted. Accordingly, the Court construes "compliance policy" as "a set for rules for determining whether a device has met a goal or objective."

[4] *See, e.g.*, U.S. Patent 9,071,518 at 19:30–59 (showing an exemplary list of the types of rules that may be selected by an administrator for determining compliance).

19. "real-time" / "real time" ('948 Patent, Claims 1, 2); "substantially real time" / "substantially real-time data" ('518 Patent, Claims 1, 10, 17)

Term	Taasera's Construction	Defendants' Construction
"real-time" / "real time"	"without intentional delay, given the processing limitations of the system"	Immediate.
"substantially real time"/ "substantially real-time data"	"without intentional delay, given the processing limitations of the system"	Indefinite.

The claims of these patents use these disputed terms as adjectives. Specifically, Claims 1 and 2 of the '948 Patent use "real time" as a modifier for "operational integrity," "behavior based events," "status indications," and "operations." *See* '948 Patent at 39:32–40:14. Claim 1 of the '518 Patent recites gathering status information of mobile devices "in a substantially real time manner." '518 Patent at 23:60–63; *see also id.* at 25:38–39 (requiring, in Claim 10, "a data store . . . that includes substantially real-time data"), 26:25–26 (requiring, in Claim 17, a data store "configured to gather substantially real-time data").

The dispute centers on the difference, if any, between "real time" and "substantially real time." In its opening brief, Taasera asserts both terms should be given the same construction.[5] Dkt. No. 256 at 32–33. Defendants counter the terms have different meanings and that "substantially real time" is an indefinite term of degree. Dkt. No. 259 at 27–29.

A. "real time," "real-time" ('948 Patent, Claims 1, 2)

Pointing to extrinsic evidence, Taasera defines "real time" as "[t]hat which occurs instantaneously, or so quickly that processing, entering, adaptation, or any other response is [at] least as

[5] In its reply, Taasera asserts "real time" should be given a "plain and ordinary meaning" construction. Dkt. No. 272 at 11–12.

fast as a triggering event or circumstance." Dkt. No. 256 at 33 (citing Wiley Elec. & Elecs. Dict.). Taasera then argues that "what is or is not 'real time' must be understood within the processing limits of the system." *Id.* "For example," says Taasera, "mobile networks may be considered 'real-time,' even if information takes several seconds (or more) to be collected from mobile devices due to network traffic." *Id.*

Defendants dispute that "real-time" means "without intentional delay," as originally proposed by Taasera, and criticize Taasera's construction as "it takes however long it takes." Dkt. No. 259 at 28. Taasera's construction, say Defendants, "is contrary to common sense and to the '948 Patent's focus on providing an immediate visibility into the integrity of the application to prevent exploits and data breaches." *Id.*

The Court agrees with Defendants. A skilled artisan would not understand "real time" as depending on the processing limitations of the system. If, for example, a video conference between two people included a 1-hour lag in communication because of the processing limits of the system, the participants would likely not consider that conference to be happening in "real-time." That said, Defendants' construction is too narrow. Nothing is absolutely immediate, even with computers, and Defendants acknowledge as much. *See* Hr'g Tr., Dkt. No. 301 at 92:1–8 (recognizing a small delay could come within a person's understanding of "immediate").

"Real time" is a well-understood concept even to a lay juror. Accordingly, the Court will give it a "plain and ordinary meaning" construction while rejecting Taasera's original requirement that "real time" depends on the processing limitations of the system or the "intentionality" of the delay.

B. "substantially real time," "substantially real-time data" ('518 Patent,

Claims 1, 10, 17)

Defendants challenge this term as an indefinite term of degree, and assert "the patent fails to provide sufficient guidance for determining scope." Dkt. No. 259 at 30. Taasera counters that the patent provides a standard for measuring the degree. *Id.* at 34 (citing '518 Patent at 10:48–65).

The Court agrees with Defendants. Notably, the '518 Patent uses both "real time" and "substantially real time" in the written description. *See, e.g.*, '518 Patent at 5:21–24 ("[B]y gathering information from each mobile device that accesses a corporate network, mobile device inventory can be monitored to provide *real-time* inventory information." (emphasis added)); *id.* at 10:44–51 (noting "message handlers 18 provide *real-time status updates* 22 to be stored and organized as records in device database 20" and that using those real-time status updates, "device database 20 can update these records and provide *substantially real-time* information about the status of mobile devices 10" (emphasis added)). This suggests some difference in scope between "real time" and "substantially real time," but Taasera points to nothing that explains where one ends and the other begins. Instead, Taasera points to an excerpt apparently unrelated to the boundaries of "substantially real-time." Dkt. No. 256 at 34 (citing '518 Patent at 10:48–65, which uses, but does not explain the scope of, "substantially real-time"). Nor does Taasera's expert's declaration help. *See* Cole Decl., Dkt. No. 256-17 ¶ 82 (providing a conclusory opinion that "this claim phrase is not indefinite, as the POSITA would understand that the word "substantially" does not render the claim indefinite" and failing to explain the demarcation between "real time" and "substantially real time"). The Court therefore concludes Defendants have carried their clear and convincing burden of showing these terms are indefinite.

20. "which includes a network analyzer, an integrity processor, an event correlation matrix, a risk correlation matrix, and a trust supervisor" ('948 Patent, Claim 1)

Taasera's Construction	Defendants' Construction
Plain and ordinary meaning.	Indefinite.

Claim 1 requires a "native computing environment which includes a network analyzer, an integrity processor, an event correlation matrix, a risk correlation matrix, and a trust supervisor." '948 Patent at 40:1–4. The dispute, however, centers only on "integrity processor" and "trust supervisor." Defendants argue these were not well-known terms of art, and that the intrinsic record fails to provide a definition of the terms. Dkt. No. 259 at 21 (citing Rubin Decl., Dkt. No. 256-25 ¶¶ 45–65). Taasera replies that the specification has clearly labeled sections for each of these components. Dkt. No. 272 at 13 (citing '948 Patent at 17:16–34, 22:26–24:67). Although Taasera does not dispute these terms were not well-known at the time of invention, it fails to set forth what it believes to be their boundaries.

"If . . . the claim term does not have an ordinary meaning, and its meaning is not clear from a plain reading of the claim, '[courts] turn to the remaining intrinsic evidence, including the written description, to aid in [the] construction of that term.'" *Power Integrations, Inc. v. Fairchild Semiconductor Int'l, Inc.*, 711 F.3d 1348, 1361 (Fed. Cir. 2013) (quoting *Telemac Cellular Corp. v. Topp Telecom, Inc.*, 247 F.3d 1316, 1326 (Fed. Cir. 2001)). "[T]he specification is always highly relevant to the claim construction analysis. Usually, it is dispositive; it is the single best guide to the meaning of a disputed term." *Vitronics Corp. v. Conceptronic,* 90 F.3d 1576, 1582 (Fed. Cir. 1996); *see also Irdeto Access, Inc. v. Echostar Satellite Corp.*, 383 F.3d 1295, 1300 (Fed. Cir. 2004) ("[A]bsent such an accepted meaning, we construe a claim term only as broadly as provided for by the patent itself. . . . The duty thus falls on the patent applicant to provide a precise definition

for the disputed term.”).

A. “integrity processor”

Taasera points to the following:

> With reference to FIGS. 6 [below] and 7A–7C, the integrity processor 630 is a functional component of the endpoint trust agent 510. The integrity processor 630 can be configured to receive integrity events 621 from the runtime monitor 620 that describes process, processor, system and binary analysis events illustrated in FIG. 7A. Applying qualifying rule expressions (illustrated in FIG. 7B) in the rulesets 631, integrity alerts 632 are generated and mapped to a cell in an event correlation matrix grid 633 (analogous to the risk correlation matrix 411 of FIG. 4 and grid 721 of FIG. 7C) to generate system level integrity warnings 634. The rows in the event correlation matrix 633 represent an application instance on the device 560 (analogous to rows in the risk correlation matrix 411 that represent a device by machine identifier). The integrity warnings 634 can be formatted as endpoint events 520 for dispatch to the system event correlator 410 of the trust orchestrator 430 for threat classification and identification and subsequent remediation.

’948 Patent at 17:16–34.

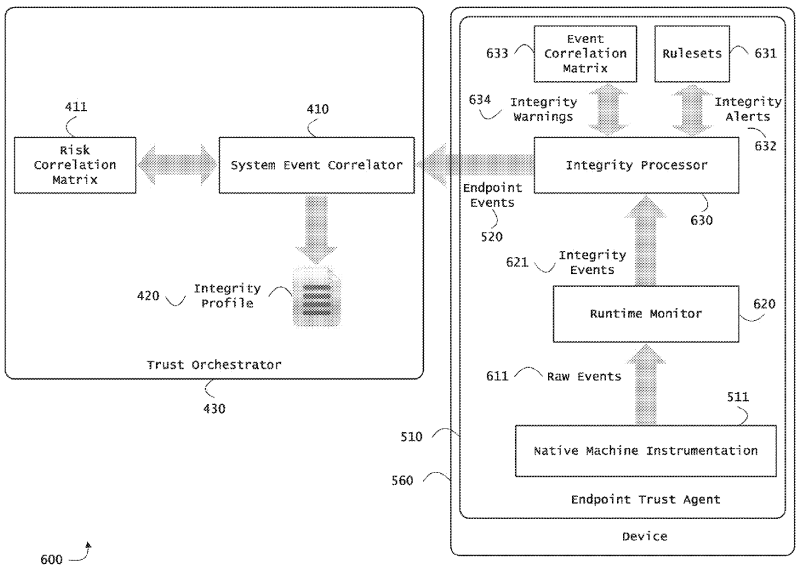

FIG. 6 of the ’948 Patent

The parties don’t dispute the lack of a well-known meaning of “integrity processor.”

Because that meaning is not clear from a plain reading of the claim, the Court turns to the only description of an "integrity processor" from the specification, quoted *supra*. Based on that excerpt, the Court construes "integrity processor" as:

> software within an endpoint trust agent that can (1) receive integrity events from a runtime monitor, (2) apply qualifying rule expressions from rulesets, (3) generate system-level integrity warnings based on an event correlation matrix, and (4) dispatch endpoint events, which are formatted from the integrity warnings, to a system event correlator external of the endpoint trust agent.

B. "trust supervisor"

Figures 8 and 11 show a "trust supervisor." Taasera points to column 22, line 26, through column 24, line 67. Dkt. No. 272 at 13. There, under the heading "Trust Supervisor," the patent explains:

> According to the exemplary embodiments of FIG. 8 and FIG. 10, the trust supervisor 846 can be configured to receive infection profiles 832 from the network analyzers 830 and integrity profiles 420 from the system event correlator 410. The trust supervisor 846 uses an event processor 1006, an event correlator 1007, a threat classifier 1008, an event correlation matrix 1013 and a rules repository 1012 to correlate events associated to a device 560 for the classification and identification of threats based on the forensic confidence of leading indicators.

'948 Patent at 22:26–36. Despite the boilerplate reference to this description being "exemplary," this first paragraph of the three columns cited by Taasera describes the "trust supervisor" generally. The Court therefore construes "trust supervisor" as:

> software that can (1) receive infection profiles from a network analyzer, (2) receive integrity profiles from a system event correlator, and (3) use an event processor, an event correlator, a threat classifier, an event correlation matrix, and a rules repository to correlate events associated with a device for the classification and identification of threats.

21. "operational integrity of the application" ('948 Patent, Claim 1)

Taasera's Construction	Defendants' Construction

Plain and ordinary meaning.	"the level of threat or contextual trustworthiness of the application"

Claim 1 recites "generating real-time behavior based events for determining the real-time operational integrity of the application." '948 Patent at 39:40–41. The parties dispute whether "operational integrity" requires identifying a threat or trustworthiness *level*. Defendants focus on the Abstract, which explains "[a] security orchestration service generates runtime operational integrity profiles representing and identifying a level of threat or contextual trustworthiness, at near real time, of subjects and applications on the instrumented target platform." Dkt. No. 259 at 22 (citing '948 Patent at [57]). Taasera stresses this language refers not to "operational integrity," but to "operational integrity *profiles*," Dkt. No. 256 at 35, and assert Defendants' construction is too narrow in light of column 12, line 55, through column 13, line 15.

That section of the patent, titled "System for Evaluating Operation Integrity of Applications," does little to explain the meaning of the disputed term. It does, however, show "application operational integrity" can be evaluated and "assessed based on rule sets." *See* '948 Patent at 12:58–61 ("FIG. 5 depicts communications between components of the application operational integrity system 500 used to evaluate application integrity based upon executable image profiles and process monitoring[.]"), 13:10–13 ("the trust agent 510 also comprises an application integrity module 516 and a resource utilization module 515 configured to assess operational integrity based on rulesets 517"). Based on that description and its distinction from the language in the Abstract, Defendants' construction is too narrow and geared toward how to *indicate* a measure of operational integrity rather than the meaning of the term itself. In other words, nothing about "operational integrity" inherently requires it to be *indicated* a certain way, such as with levels. As such, the Court will give this term a "plain and ordinary meaning" construction.

22. "an event correlation matrix" ('948 Patent, Claim 1)

Taasera's Construction	Defendants' Construction
Plain and ordinary meaning.	"A matrix which maps integrity warnings to endpoint events, and the rows or columns represent an application instance on the monitored system."

As noted *supra*, Claim 1 requires a "native computing environment which includes . . . an event correlation matrix." '948 Patent at 40:1–4. Defendants assert this term has no "plain meaning" outside the context of the '948 Patent and point to the description of Figure 6 for its construction. Dkt. No. 259 at 24–25. Taasera does not dispute the term has no "plain meaning," but asserts Defendants' construction adds nothing further to what a skilled artisan would understand from reading the specification. Dkt. No. 272 at 14.

Given there is no "plain meaning," the Court looks to the specification for aid in how to construe the term. *See Irdeto Access, Inc.*, 383 F.3d at 1300 ("[A]bsent such an accepted meaning, we construe a claim term only as broadly as provided for by the patent itself. . . . The duty thus falls on the patent applicant to provide a precise definition for the disputed term."). As to the first part of Defendants' construction—what is being correlated—Defendants have the better position. Referring to FIG. 6, which shows "an exemplary correlation system," the patent explains:

> The runtime monitor 620 is further configured to raise integrity events 621 to an integrity processor 630 that may apply rulesets 631 to generate integrity alerts 632 and generate or populate an event correlation matrix 633 *that maps integrity warnings 634 to endpoint events 520 for dispatch to a trust orchestrator 430.*

'948 Patent at 15:14–19 (emphasis added). The first part of Defendants' construction aligns with that explanation.

As for the second part of Defendants' construction, referring to "row" and "columns" in this context is overly restrictive and potentially confusing to a jury given that the "matrix" is stored

in a computer as binary data. That is, a matrix is not stored in memory in literal rows or columns, as might be used to represent the matrix on a display or paper as a grid. Regarding what the rows or columns represent—Defendants propose "an application instance"—the surrounding claim language is sufficiently informative on that question.

In sum, the Court adopts the first part of Defendants' construction, but rejects the second. Specifically, the Court construes "an event correlation matrix" as "a matrix which maps integrity warnings to endpoint events."

23. "a risk correlation matrix" ('948 Patent, Claim 1)

Taasera's Construction	Defendants' Construction
Plain and ordinary meaning.	"A matrix which maps endpoint events to system warning classes, system warnings, and/or integrity warnings, and the rows or columns represent a machine identifier or application instance identifier of the monitored system."

The parties' arguments for this term track their arguments for "an event correlation matrix." *See* Dkt. No. 256 at 37 (arguing the disclosures of such a matrix are exemplary and there is no lexicography); Dkt. No. 259 at 25–26 (arguing there is no "plain meaning" for the term); Dkt. No. 272 at 14–15 (asserting "Defendants' proposed construction adds nothing further to what a POSITA would receive from reading the specification").

The patent provides the following explanation under the heading "Risk Correlation Matrix":

> In embodiments, the risk correlation matrix 411 depicted in FIGS. 4-6 and the risk correlation matrix 721 described with reference to 7A–7C are embodied as grids that represent an exemplary dynamic model of measurement and identification based on clustering and classification of independent endpoint events (alerts) to generate system warnings that may be mapped to warning categories or classes. Additional details of the risk correlation matrices 411 and 721 are provided below with reference to FIGS. 6, 7B, and 7C.

'948 Patent at 12:46–54.

In its reply, Taasera concedes Defendants' construction "adds no further understanding to what a POSITA would receive from reading the specification," Dkt. No. 272 at 14–15, thus suggesting Defendants' construction is unnecessary but correct. Thus, for the same reasons noted *supra* for "an event correlation matrix," the Court adopts the first part of Defendants' construction and rejects the second. Specifically, the Court construes "a risk correlation matrix" as "a matrix which maps endpoint events to system warning classes, system warnings, and/or integrity warnings."

24. "correlating, by the event and risk correlation matrix" ('948 Patent, Claim 1)

Taasera's Construction	Defendants' Construction
Plain and ordinary meaning.	Indefinite.

This phrase appears as part of the penultimate limitation of Claim 1, which recites "*correlating, by the event and risk correlation matrix*, threat classifications based on the temporal sequence of the generated real-time behavior based events[.]" '948 Patent at 40:5–7. Those "real-time behavior based events" are generated "on the native computing environment which includes a network analyzer, an integrity processor, *an event correlation matrix, a risk correlation matrix,* and a trust supervisor[.]" *Id.* at 39:40–40:4 (emphasis added).

Defendants allege the disputed phrase is ambiguous and could as many as four meanings.[6] Dkt. No. 259 at 27. It stresses the phrase "event and risk correlation matrix" is not in the

[6] Those four meanings are: (1) "correlating, by the event ~~and risk~~ correlation matrix"; (2) "correlating, by the ~~event and~~ risk correlation matrix"; (3) "correlating, by the event <u>correlation matrix</u> and <u>the</u> risk correlation matrix"; or (4) "correlating, by ~~the~~ <u>an</u> event and risk correlation matrix." Dkt. No. 259 at 27.

specification and was not a well-known term of art. *Id.* Taasera, however, points to the claim's earlier recitation of "event correlation matrix" and "risk correlation matrix," and assert those matrices are to what this phrase refers. Dkt. No. 256 at 38. In response, Defendants accuse Taasera of asking for correction, which they say is inappropriate here. Dkt. No. 259 at 26.

Correction is not necessary. A skilled artisan would understand "the event and risk correlation matrix" refers back to the previously recited matrices. The specification confirms that conclusion by describing the use of the event correlation matrix and risk correlation matrix by event and risk correlators, respectively. *See* '948 Patent at 22:66–23:3 (describing an event correlator that maps alerts using an event correlation matrix); *id.* at 25:51–54 (describing a risk correlator configured to use a risk correlation matrix). Accordingly, the Court construes this term as "correlating, by the event correlation matrix and the risk correlation matrix."

27. "initiating . . . at least one action" / "initiate an action" ('518 Patent, Claims 1, 10, 17)

Taasera's Construction	Defendants' Construction
Plain and ordinary meaning.	Initiating: "automatically causing to being in real time"

The '518 Patent concerns "remote management of mobile devices at a server using rules based actions." '518 Patent at 1:16–17. The patent, which has a 2011 effective filing date, describes a "general trend in corporate network management" of employees using mobile devices, such as smartphones and tablets, to access corporate networks. *Id.* at 1:23–31. This understandably presented security challenges to network operators because of, for example, the different types of devices and operating systems that might be used. *See id.* at 1:32–62. Since network managers rarely have the expertise to handle all possible platforms and devices, the patent purports to teach how to "simplify the management and configuration of these mobile devices by an IT manager."

Id. at 1:62–2:2.

The patent generally describes the invention as a "server-based rules-based action framework that gathers status and configuration information about the mobile devices to be used. '518 Patent at [57]. The server processes that status information and automatically responds to changes based on administrator-selected rules. *Id.* Examples of rules that might be enforced include limited access to the network based on the versions of operating systems, allowing or prohibiting access based on geo-fencing, and requiring encryption for access. *See id.* at 19:30–59 (Table 1).

This dispute concerns method Claims 1 and 10 and system Claim 17. Claim 1 recites:

> 1. A method for providing device management services comprising steps of:
>
> gathering, at a server, from a network, status information related to an operating state of each of a plurality of mobile devices, each mobile device utilizing one of a plurality of different mobile operating systems, wherein the status information for each mobile device is gathered from a plurality of sources including each mobile device in a substantially real time manner;
>
> formatting the status information for each of the mobile devices, such that rules can be substantially uniformly applied to the status information across the plurality of different mobile operating systems; storing the status information in an electronic database accessible to the server;
>
> evaluating, at the server, a first compliance state of each of the mobile devices from the status information using a plurality of administrator-defined rules, wherein each of the rules applies to mobile devices regardless of mobile operating system; and
>
> *initiating, at the server, at least one action defined by the administrator-defined rules in response to the step of evaluating,*
>
> wherein the step of evaluating is performed by the server automatically, in response to changes in the status information in the electronic database.

'518 Patent at 23:55–24:57 (emphasis added). Notably, while Claim 1 only recites "initiating" an action, the other claims at issue require the "initiating" to be "automatic." *See id.* at 26:22–24 (reciting, in Claim 10, the step of "automatically initiating, at the server, an action defined by the at least one of a plurality of administrator defined rules if [an] attribute is out of compliance"); *id.* at 26:6–13 (reciting, in Claim 17, "[a] system for providing device management services" comprising computer readable instructions configured to "automatically initiate an action").

Defendants argue the recited "initiating" must be done both "automatically" and "in real time." First, say Defendants, the Summary and Abstract show the "initiating" step is done automatically based on predetermined rules and amounts to "causing" the recited action. Dkt. No. 259 at 33 (citing '518 Patent at [57], 2:15–17, 2:62–65, 3:27–33, 4:41–4). Second, the applicants disclaimed a broader construction during prosecution. *Id.* (citing Am. & Req. for Reconsideration, Dkt. No. 259-7 at 7; Am. & Req. for Reconsideration, Dkt. No. 259-8 at 7). In its reply, Taasera makes a claim differentiation argument based on Claims 10 and 17, and argues "in real time" is not necessary to understand "initiating." Dkt. No. 272 at 15.

A. The Prosecution History

Defendants' prosecution-history argument concerns the applicant's attempts to distinguish the claims over Thomas in May 2014 and February 2015 responses to USPTO rejections.

1. The May 2014 Response

In the May 2014 response, the applicants amended then-pending Claims 1, 10, and 17 to read:

> 1. A method for providing device management services comprising steps of:

gathering, from a network, status information related to each of a plurality of mobile devices, each mobile device utilizing one of a plurality of different mobile operating systems;

storing the status information in an electronic database accessible to a server;

evaluating at the server a first compliance state of at least one of the mobile devices from the status information using a plurality of administrator-defined rules; and

initiating at least one action defined by the [[user]] administrator-defined rules in response to the step of evaluating,

wherein the step of ~~evaluation~~ evaluating is performed by [[a]] the server automatically, in response to changes in the status information in the electronic database.

* * *

10. A method for providing device management services comprising steps of:

making a determination that an attribute of a mobile device has changed;

evaluating the attribute, by a processor at a server in response to the determination, to determine if it triggers at least one of a plurality of administrator defined rules, which indicates that the attribute is out of compliance; and

automatically initiating, at the server, an action defined by the at least one of a plurality of administrator defined rules if the attribute is out of compliance,

wherein the attribute is stored in a data store accessible to the server, that includes substantially real-time data, gathered from a network, pertaining to a plurality of attributes of

a plurality of mobile devices, comprising devices having different mobile operating systems.

* * *

17. A system for providing device management services comprising:

> a data store configured to be accessible by a server;
>
> a CPU of the server coupled to a memory configured to execute computer readable instructions;
>
> computer readable instructions for execution on the server comprising instructions configured to:
>
> automatically make a determination that an attribute of a mobile device stored in the data store has changed;
>
> evaluate the attribute, in response to the determination, to determine if it triggers at least one of a plurality of administrator defined rules, which indicates that the attribute is out of compliance; and
>
> automatically initiate an action defined by the at least one of a plurality of administrator defined rules if the attribute is out of compliance,
>
> wherein the data store is configured to ~~include~~ gather, from a network, substantially real-time data pertaining to a plurality of attributes of a plurality of mobile devices, comprising devices having different mobile operating systems.

Am. & Req. for Reconsideration, Dkt. No. 259-7 at 2–5. In connection with those amendments, the applicants argued Thomas fails to teach

> a "step of evaluating is performed by the server automatically, in response to changes in the status information in the electronic database." [Claim 1, exemplary] By responding automatically at the server to changes in status information in a database, the claimed invention is fundamentally different from the system taught be Thomas . . . .

Id. at 7 (brackets in original).

Neither these amendments nor arguments support any disclaimer concerning the "initiating" limitations. For one, the applicants did not make arguments for patentability based on the alleged "real time" character of a limitation. Nor did they distinguish Thomas based on the "initiating" step. Instead, they distinguished the claims based on the "evaluating" step using the express

language of the claims at the time. That is not clear and unmistakable disclaimer that narrows the ordinary meaning of terms.

2. The February 2015 Response

In the February 2015 response, the applicants further amended the claims to read:

> 1. A method for providing device management services comprising steps of:
>
> gathering, at a server, from a network, status information related to an operating state of each of a plurality of mobile devices, each mobile device utilizing one of a plurality of different mobile operating systems, wherein the status information for each mobile device is gathered from a plurality of sources including each mobile device in a substantially real time manner;
>
> formatting the status information for each of the mobile devices, such that rules can be substantially uniformly applied to the status information across the plurality of different mobile operating systems;
>
> storing the status information in an electronic database accessible to [[a]] the server, evaluating, at the server, a first compliance state of ~~at least one~~ each of the mobile devices from the status information using a plurality of administrator-defined rules, wherein each of the rules applies to mobile devices regardless of mobile operating system; and
>
> initiating, at the server, at least one action defined by the administrator-defined rules in response to the step of evaluating,
>
> wherein the step of evaluating is performed by the server automatically, in response to changes in the status information in the electronic database.
>
> * * *
>
> 10. A method for providing device management services comprising steps of:
>
> receiving, at a server, a plurality of attributes related to an operating state of each of a plurality of mobile devices, each

mobile device utilizing one of a plurality of different mobile operating systems, wherein the attributes for each mobile device are gathered from a plurality of sources including each mobile device;

making a determination that an attribute of a mobile device has changed;

evaluating the attribute, by a processor at [[a]] the server in response to the determination, to determine if [[it]] the attribute change triggers at least one of a plurality of administrator-defined rules, which indicates that the attribute is out of compliance, wherein each of the rules applies to mobile devices regardless of mobile operating system; and

automatically initiating, at the server, an action defined by the at least one of a plurality of administrator defined rules if the attribute is out of compliance,

wherein the attribute is stored in a data store accessible to the server, that includes substantially real-time data, gathered from the plurality of sources on a network[[,]] and pertaining to a plurality of attributes of a plurality of mobile devices, ~~comprising devices having~~ wherein the substantially real-time data is formatted, such that the rules can be substantially uniformly applied to the plurality of attributes across the plurality of different mobile operating systems.

* * *

17. A system for providing device management services comprising:

a data store configured to be accessible by a server;

a CPU of the server coupled to a memory configured to execute computer readable instructions;

computer readable instructions for execution on the server comprising instructions configured to:

automatically make a determination that an attribute of a mobile device stored in the data store has changed;

evaluate the attribute, in response to the determination, to

> determine if it triggers at least one of a plurality of administrator defined rules, which indicates that the attribute is out of compliance; and
>
> automatically initiate an action defined by the at least one of a plurality of administrator defined rules if the attribute is out of compliance; and
>
> wherein the data store is configured to ~~include~~ gather, from a network, substantially real-time data pertaining to a plurality of attributes of a plurality of mobile devices, comprising devices having different mobile operating systems.

Am. & Req. for Reconsideration, Dkt. No. 259-8 at 2–5. In their remarks concerning these amendments, the applicants argued

> Thomas teaches a system that provides for centralized policy updates to client facilities, to allow these client facilities to implement and/or participate in the application of the policies. Client facilities then use the security information received from the server to ensure compliance. For example, software can be provided to perform disk scans of client files on disk, or as they are sent to or from the client. Thomas further teaches URI filtering and monitoring, monitoring behavior before allowing execution of software, monitoring outgoing files. *These are all processes that necessarily take place at the client. In contrast, the rule enforcement and actions in the pending claims take place on the server, in response to real-time status information received from mobile devices.*

Id. at 7 (citations omitted).

This shows Applicants made the key distinction that the steps of the claims were performed *at the server* rather than *at the client*. The applicants also noted that, relative to Thomas, the rule enforcement and actions take place "in response to real-time status information received from mobile devices." That argument relates to the "real-time" character of the status information rather than the "real-time" character of the response. Again, this prosecution history is not sufficiently clear to be a disclaimer of scope as urged by Defendants.

B. The Specification

Defendants' specification-based argument concerns only the "automatically" portion of its proposed construction. They cite five excerpts in particular.

(1) "Each rule includes conditions and an action pre-selected by an administrator, which *will automatically be initiated*," '518 Patent at 2:15–17;

(2) "If the attribute is out of compliance, and the method *automatically initiates*, by the server, an action defined by the at least one of a plurality of administrator defined rules," *id.* at 2:62–65;

(3) "The instructions are further configured to evaluate the attribute, in response to the determination, to determine if it triggers at least one of a plurality of administrator defined rules, which indicates that the attribute is out of compliance and *automatically initiate an action* defined by the at least one of a plurality of administrator defined rules if the attribute is out of compliance," *id.* at 3:27–33;

(4) "The present invention solves many issues of managing a plurality of mobile devices by providing a server-based rules system for gathering status and configuration information related to each mobile device being managed and applying administrator-defined rules to *automatically carry out actions* to assist in managing the devices," *id.* at 4:41–46; and

(5) "At the server, software processes monitor status information and respond automatically to changes, causing administrator-selected rules to be evaluated to determine if *an action should automatically be initiated*," *id.* at [57].

But as noted in the description of the prosecution history, Claims 10 and 17 already expressly require the "initiating" step to be "automatic." And regarding Claim 1, "a claim need not read on every described embodiment." *Kruse Tech. P'ship v. Volkswagen AG*, 544 Fed. Appx. 943, 951 (Fed. Cir. 2013). Thus, the Court holds there is no clear and unmistakable disclaimer of scope from the ordinary meaning of the claim language based on the specification.

The Court will give this phrase a "plain and ordinary meaning" construction.

IV. CONCLUSION

No.	Term	The Court's Construction
2	"if the new program is validated, permitting the new program to continue loading and to execute in connection with the computing device" ('137 Patent, Claims 6, 14, 25)	Plain and ordinary meaning
3	"an execution module coupled to the detection module and operable for monitoring, at the operating system kernel of the computing device, the program in response to the trigger intercepted by the detection module" ('137 Patent, Claim 1)	**§ 112 ¶ 6 Structure:** a processor programmed to (1) detect network activity, an attempt to open a file, or an attempt to change the registry from a system call hook triggered by the kernel; and (2) communicate the system call hook to the associated in-kernel behavior monitor using a component interface
4	"network administration traffic" ('356 Patent, Claims 1, 2, 9, 10, 13, 14, 17, 18)	"packets having an IP protocol and address that match an IP protocol and address in a list of IP protocols and addresses of known administrators"
	"[third/fourth] program instructions to determine if the packet is network administration traffic" ('356 Patent, Claims 1, 9, 10, 13, 17)	**§ 112 ¶ 6 Structure:** "a processor programmed to (1) determine the IP protocol and IP address of the packet; (2) compare the determined IP protocol and IP address of the packet to a list of IP protocols and addresses of known administrators; and (3) determine whether the comparison results in a match"
5	"attestation" ('441 Patent, Claims 1, 2, 3, 5, 7; '616 Patent, Claim 1)	Plain and ordinary meaning
6	"runtime" ('616 Patent, Claim 1)	"the time when operations are performed on the device, including running applications and device functions" The preamble is limiting.

No.	Term	The Court's Construction
7	"network trust agent" ('616 Patent, Claim 1,)	"software that sends policies to a network access point to block or divert traffic flows for user devices based on systems warnings associated with those devices" The preamble is limiting.
8	"at runtime receiving . . . a runtime execution context indicating attributes of the application at runtime, wherein the attributes comprise one or more executable file binaries of the application and loaded components of the application" ('441 Patent, Claims 1, 4)	Plain and ordinary meaning
9	"a security context providing security information about the application" ('441 Patent, Claims 1, 4, 5)	Plain and ordinary meaning
10	"an application artifact" ('441 Patent, Claim 2)	"data identifying one or more attribute value assertions that describe the application runtime execution context"
11	"introspective security context" ('441 Patent, Claims 4, 5)	"a security evaluation based on information and measurements [of the remote computing platform] sampled over a period of time"
12	"the application of the restriction of the user's transaction" ('441 Patent, Claim 11)	Indefinite.
13	"return URL" ('419 Patent, Claims 1, 3, 4, 6, 7, 9, 10, 12, 13, 15, 16, 19; '634 Patent, Claims 1, 3, 4, 5, 6, 8; '251 Patent, Claims 1, 2, 3, 4, 5, 6, 8, 9, 10; '453 Patent, Claims 1, 5, 6, 10, 11, 12, 16, 17, 18)	Plain and ordinary meaning

No.	Term	The Court's Construction
14	"evaluating[, by the computer,] the URL to determine whether encryption of [none, part, or all of]the URL is required" ('419 Patent, Claims 1, 4, 10, 13, 17; '634 Patent, Claims 1, 4) "determining, by the computer, whether encryption is required for none, part, or all of a return URL" / "determining[, by the computer,] [whether/that] encryption of [a/the] return URL [of the requested resource] is required" / "determining by the computer, [whether/that] encryption of the contained URL [is/is not] required" / "determine that encryption of the URL is not required" ('419 Patent, Claims 1, 4, 13, 19; '634 Patent, Claims 1, 4; '251 Patent, Claims 1, 2, 3, 4, 5, 6, 8, 9, 10; '453 Patent, Claims 1, 4, 6–18)	Plain and ordinary meaning
16	"determining[, by the computer,] whether the URL of the requested resource is required" ('419 Patent, Claims 2, 11; '634 Patent, Claim 2)	"determining[, by the computer,] whether encryption of the URL of the requested resource is required"
17	"compliance state of the endpoint" ('038 Patent, Claims 1, 12, 23; '997 Patent, Claims 1, 11, 21; '918 Patent, Claims 1, 9, 17)	"the state of being either in or not in compliance"
18	"compliance polic[y/ies]" ('038 Patent, Claims 1, 12, 23; '997 Patent, Claims 1, 11, 21; '918 Patent, Claims 1, 9, 17)	"a set for rules for determining whether a device has met a goal or objective"

No.	Term	The Court's Construction
19	"real-time" / "real time" ('948 Patent, Claims 1, 2)	Plain and ordinary meaning.
	"substantially real time"/ "substantially real-time data" ('518 Patent Claims 1, 10, 17)	Indefinite.
20	"which includes a network analyzer, an integrity processor, an event correlation matrix, a risk correlation matrix, and a trust supervisor" ('948 Patent, Claim 1)	**integrity supervisor:** a software module within an endpoint trust agent that can (1) receive integrity events from a runtime monitor, (2) apply qualifying rule expressions from rulesets, (3) generate system-level integrity warnings based on an event correlation matrix, and (4) dispatch endpoint events, which are formatted from the integrity warnings, to a system event correlator external of the endpoint trust agent. **trust supervisor:** "a software module that can (1) receive infection profiles from a network analyzer (2) receive integrity profiles from a system event correlator; (3) use an event processor, an event correlator, a threat classifier, an event correlation matrix, and a rules repository to correlate events associated with a device for the classification and identification of threats"
21	"operational integrity of the application" ('948 Patent, Claim 1)	Plain and ordinary meaning.
22	"an event correlation matrix" ('948 Patent, Claim 1)	"a matrix which maps integrity warnings to endpoint events"
23	"a risk correlation matrix" ('948 Patent, Claim 1)	"a matrix which maps endpoint events to system warning classes, system warnings, and/or integrity warnings"
24	"correlating, by the event and risk correlation matrix" ('948 Patent, Claim 1)	"correlating, by the event correlation matrix and the risk correlation matrix"

No.	Term	The Court's Construction
27	"initiating . . . at least one action" / "initiate an action" ('518 Patent, Claims 1, 10, 17)	Plain and ordinary meaning.

The Court **ORDERS** each party not to refer, directly or indirectly, to its own or any other party's claim-construction positions in the presence of the jury. Likewise, the Court **ORDERS** the parties to refrain from mentioning any part of this opinion, other than the actual positions adopted by the Court, in the presence of the jury. Neither party may take a position before the jury that contradicts the Court's reasoning in this opinion. Any reference to claim construction proceedings is limited to informing the jury of the positions adopted by the Court.

So ORDERED and SIGNED this 13th day of December, 2023.

RODNEY GILSTRAP
UNITED STATES DISTRICT JUDGE